TCEPE

MANUAL DE CONTROLES DE SEGURANÇA

WEB, E-MAIL E DNS

WEB | E-MAIL | DNS

Boas práticas e requisitos essenciais para garantir segurança, confiabilidade e conformidade.

Sumário

Introdução

O que é este manual?

Este documento é um guia prático de segurança digital desenvolvido para apoiar gestores públicos, equipes de tecnologia da informação e responsáveis por domínios institucionais na compreensão e implementação dos controles verificados pela plataforma TOP (Teste os Padrões), mantida pelo NIC.br — o Núcleo de Informação e Coordenação do Ponto BR, entidade responsável pelo registro de domínios ".br" e pela operação de infraestruturas críticas da internet brasileira.

Este manual detalha as verificações técnicas executadas automaticamente pela plataforma TOP em domínios brasileiros. O documento define cada controle, justifica sua relevância, apresenta os riscos associados à sua ausência e orienta os procedimentos para sua implementação.

O que é o TOP?

O TOP é uma plataforma gratuita do NIC.br que realiza testes automatizados de segurança e boas práticas em domínios ".br", avaliando três dimensões fundamentais da presença digital institucional:

Dimensão	O que é avaliado
Presença Web	Segurança do site: HTTPS, TLS, cabeçalhos de segurança, DNSSEC.
E-mail	Autenticação de mensagens: SPF, DKIM, DMARC, DANE, STARTTLS.
Conectividade	Modernização de endereçamento: suporte a IPv6 em toda a cadeia de serviços.

Os resultados são públicos e acessíveis a qualquer pessoa em top.nic.br, permitindo que cidadãos, gestores e equipes técnicas consultem a situação de segurança de qualquer domínio ".br" a qualquer momento.

Por que esses controles foram escolhidos?

Os testes do TOP não foram definidos arbitrariamente. Eles refletem um conjunto criterioso de boas práticas internacionais consolidadas, derivadas de padrões técnicos reconhecidos globalmente, como as normas do NIST (Instituto Nacional de Padrões e Tecnologia dos Estados Unidos), as RFCs (Request for Comments — especificações técnicas da internet publicadas pela IETF, a Internet Engineering Task Force), as diretrizes da OWASP (Open Worldwide Application Security Project — fundação sem fins lucrativos dedicada a melhorar a segurança de softwares) e as recomendações de organismos como o NCSC-NL (Centro Nacional de Segurança Cibernética dos Países Baixos) e a Internet Society (organização global sem fins lucrativos dedicada a garantir que a internet permaneça aberta, segura e confiável).

Trata-se de controles que:

- Estão ao alcance de qualquer organização com domínio ".br", independentemente do tamanho ou orçamento;
- Podem ser verificados externamente, sem necessidade de acesso ao ambiente interno da organização;
- Produzem impacto imediato e mensurável na segurança dos usuários e serviços do domínio;
- Representam o estado da arte em segurança de infraestrutura web e de e-mail para a internet pública.

Para quem é este manual?

Este manual foi elaborado para diferentes perfis de leitores, cada um com necessidades distintas:

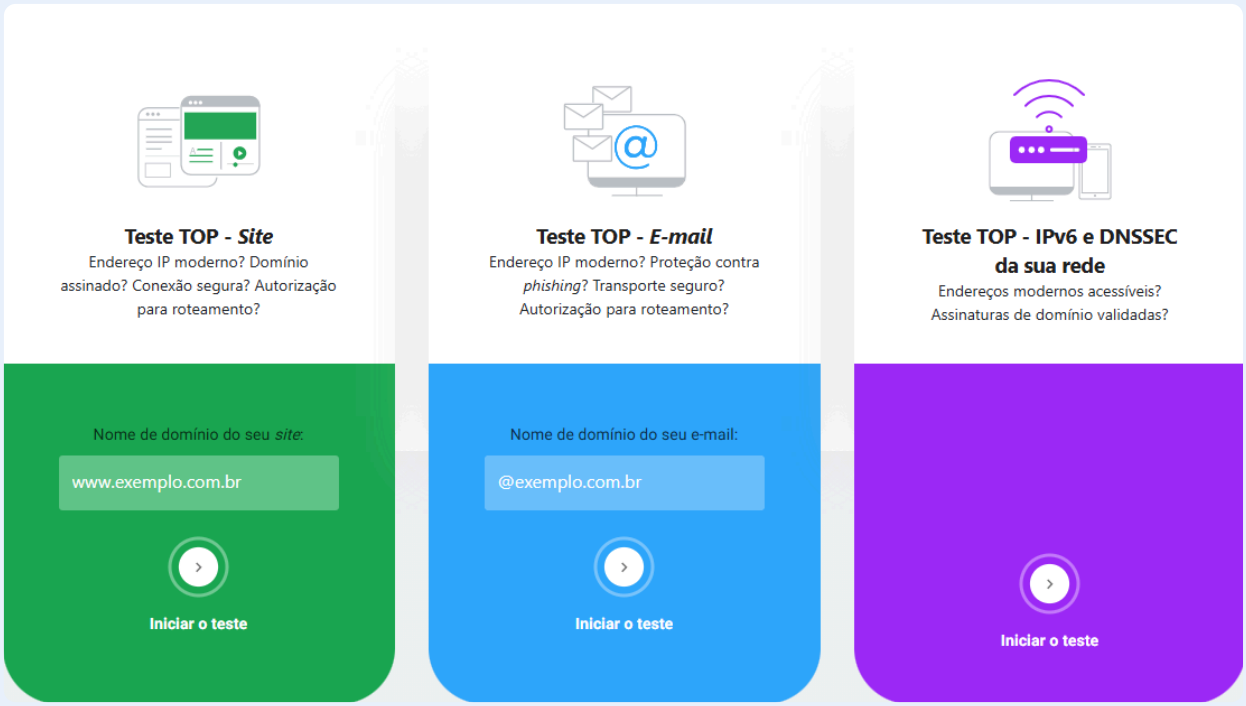
- **Gestores e tomadores de decisão** encontrarão, em cada controle, uma estimativa clara de custos financeiros, custos operacionais e benefícios esperados, permitindo priorizar ações com base em critérios objetivos de custo-benefício.
- **Equipes técnicas de TI** encontrarão orientações práticas de implementação, referências às especificações técnicas pertinentes e ferramentas recomendadas para verificação e diagnóstico.
- **Órgãos e instâncias de controle interno** encontrarão uma base estruturada para avaliação do nível de maturidade em segurança digital de domínios institucionais, alinhada aos padrões internacionais referenciados.

Como os relatórios do TOP são gerados?

A plataforma TOP realiza testes **não intrusivos e completamente externos**, ou seja, sem qualquer acesso privilegiado ao ambiente da organização avaliada. Todos os testes são executados a partir de consultas e conexões que qualquer usuário da internet poderia realizar, utilizando exclusivamente informações públicas do domínio.

Os testes são organizados em três módulos:

Figura 01 — Testes disponibilizados pela plataforma TOP



Fonte: https://top.nic.br

- **Teste de Site** (top.nic.br/test-site): avalia a segurança da presença web do domínio, verificando a disponibilidade e configuração do HTTPS, a qualidade da implementação do TLS e a presença de cabeçalhos de segurança HTTP.
- **Teste de E-mail** (top.nic.br/test-mail): avalia a infraestrutura de e-mail do domínio, verificando a existência e a configuração correta dos mecanismos de autenticação SPF, DKIM e DMARC, além da presença de criptografia no transporte de mensagens via STARTTLS e DANE.
- **Teste de Conexão** (top.nic.br/test-connection): avalia a modernização do endereçamento de rede, verificando se toda a cadeia de serviços do domínio — servidores de nomes, servidor web e

servidores de e-mail — está acessível via IPv6, em paridade com o IPv4.

Os resultados são apresentados de forma consolidada em um relatório público, identificado por uma URL única, que pode ser compartilhado, arquivado e consultado a qualquer momento. Cada item avaliado recebe uma indicação clara de conformidade, permitindo identificar rapidamente quais controles estão implementados e quais demandam atenção.

> 💡 **Como consultar:** acesse top.nic.br, insira o domínio que deseja avaliar e escolha o módulo de teste desejado. O relatório é gerado em segundos e não requer cadastro ou autenticação.

Como usar este manual?

O manual está organizado seguindo a mesma estrutura dos testes do TOP, dividida em sete seções principais:

Seção	Conteúdo
1. DNSSEC	Autenticidade e integridade das respostas DNS.
2. HTTPS e HSTS	Segurança básica da comunicação web.
3. TLS	Configuração avançada do protocolo de criptografia.
4. Cabeçalhos de Segurança HTTP	Proteções adicionais no navegador do usuário.
5. Autenticação de E-mail	SPF, DKIM, DMARC, STARTTLS e DANE.
6. Governança do E-mail Institucional	Políticas de uso, responsabilidades e conformidade na gestão do e-mail institucional.
7. Modernização de Endereçamento	Suporte a IPv6 em toda a cadeia de serviços.

A integração entre a plataforma TOP e este manual foi concebida para formar um ciclo contínuo de melhoria: o relatório do TOP aponta o que precisa ser corrigido; o manual contextualiza o risco,

subsidia a decisão de priorização e orienta tecnicamente a implementação da correção. A figura a seguir ilustra esse fluxo.

Figura 02 — Caminho recomendado de uso da ferramenta TOP

O caminho recomendado de uso é:

1. **Consultar o relatório do TOP** para o domínio institucional em top.nic.br;
2. **Identificar os controles com resultado negativo** no relatório;
3. **Localizar o controle correspondente neste manual** para compreender o risco associado e as orientações de correção;
4. **Priorizar as ações de melhoria** com base nas estimativas de custo e benefício apresentadas em cada controle, considerando a capacidade técnica e operacional da organização;
5. **Retestar o domínio no TOP** após a implementação para confirmar a conformidade.

Custos e priorização

Um dos resultados mais importantes da análise sistemática dos controles avaliados pelo TOP é a constatação de que a maioria deles tem custo financeiro nulo ou baixo. Muitos se

resumem a ajustes de configuração em sistemas já existentes, sem necessidade de aquisição de produtos, licenças ou infraestrutura adicional.

Os maiores custos associados aos controles deste manual são, em geral, operacionais: exigem conhecimento técnico qualificado, planejamento cuidadoso e atenção contínua, especialmente em relação à validade e à rotação de chaves criptográficas. Contudo, esses custos operacionais são amplamente superados pelos benefícios obtidos, considerando que os ataques mitigados por esses controles podem causar danos financeiros, reputacionais e operacionais de grande magnitude a organizações públicas e aos cidadãos que dependem de seus serviços.

A mensagem central deste manual é direta: a maioria das vulnerabilidades identificadas pelo TOP pode ser corrigida sem investimento financeiro significativo. O principal obstáculo não é o custo, mas a ausência de conhecimento sobre o problema e sobre como resolvê-lo. É exatamente esse obstáculo que este manual se propõe a remover.

1. Existência e Validade de DNSSEC

O DNSSEC compreende um conjunto de extensões de segurança para a proteção do DNS, sistema responsável pela resolução de nomes de domínio (ex.: meusite.gov.br) em endereços IP (ex.: 200.148.12.188). O protocolo assegura a autenticidade e a integridade das respostas DNS, certificando que os dados são originados do servidor autoritativo — o servidor oficialmente responsável por responder pelo domínio — e não sofreram adulteração durante a transmissão.

ℹ️ **Importante:** o DNSSEC não garante confidencialidade nem disponibilidade, apenas autenticidade e integridade dos dados DNS.

1.1 Existência de DNSSEC

Como é verificado?

O teste verifica se o registro `SOA` (Start of Authority) do domínio possui assinatura DNSSEC, confirmando que o domínio está protegido contra manipulação de respostas DNS.

Riscos da ausência

Sem DNSSEC, o domínio fica vulnerável a ataques que manipulam respostas DNS para enganar usuários:

Risco	O que acontece
Envenenamento de cache	O cache do servidor DNS é corrompido com dados falsos, redirecionando usuários a sites maliciosos sem que percebam.
Man-in-the-Middle (MitM)	Um invasor pode posicionar-se entre o usuário e o serviço legítimo por meio da falsificação de respostas DNS. O DNSSEC não impede a interceptação do tráfego em si, mas dificulta ataques em que o redirecionamento inicial depende da adulteração das respostas DNS.
Sequestro de domínio	O invasor assume o controle do domínio alterando seus registros DNS, redirecionando todos os serviços vinculados ao domínio.

Risco	O que acontece
DNS Spoofing	Usuários são redirecionados a sites falsos, visualmente idênticos ao legítimo, para roubo de credenciais ou dados.

Obrigatoriedade e recomendação

A implementação é obrigatória para os domínios: '.B.BR', '.DEF.BR', '.JUS.BR', '.LEG.BR', '.MP.BR' e '.TC.BR'. É recomendada pelo registro.br (https://registro.br/tecnologia/dnssec/dns-e-dnssec/) para todos os demais domínios brasileiros.

Como resolver?

- Verificar se o registrador ou provedor DNS atual oferece suporte a DNSSEC;
- Ativar o DNSSEC no painel de controle do registrador ou provedor DNS;
- Gerar o par de chaves criptográficas (KSK e ZSK) e cadastrar o registro `DS` correspondente junto ao registro do domínio — ou seja, na zona-mãe, e não na zona do próprio domínio. Para domínios ".br", esse cadastro é feito no painel de controle do registro.br;
- Em infraestruturas DNS próprias, considerar o uso de HSM (Hardware Security Module) para armazenamento seguro das chaves.

📚 Referências e documentação oficial

Referência	Descrição	URL
Registro.br	Guia oficial de ativação do DNSSEC para domínios .br.	ftp.registro.br/pub/doc/configuracao_dnssec_dominio.pdf
Registro.br	Tutoriais do funcionamento e configuração do DNSSEC	registro.br/tecnologia/dnssec/tutoriais/conceitos-detalhados/
NIST SP 800-81-3	Diretrizes para implantação segura de DNSSEC, incluindo recomendações sobre validade e rotação de chaves	csrc.nist.gov/pubs/sp/800/81/r3/final

> 💡 **Dica:** para domínios registrados diretamente no registro.br, a ativação do DNSSEC pode ser feita diretamente pelo painel de controle em poucos cliques, sem necessidade de configuração técnica avançada. O registro.br oferece suporte a DNSSEC sem custo adicional sobre o valor anual do domínio.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo a Médio

- A maioria dos registradores de domínio brasileiros, incluindo o registro.br, oferece suporte ao DNSSEC **sem custo adicional** sobre o valor anual do domínio.
- Caso o provedor DNS atual não suporte DNSSEC, pode ser necessário **migrar para um provedor compatível**, com custos variando entre gratuito (ex.: Cloudflare) e planos pagos de algumas centenas de reais por ano.
- Órgãos públicos devem priorizar provedores com os quais seja possível estabelecer **vínculo contratual formal, com garantias de SLA e conformidade com a LGPD**, observando as normas de contratação pública aplicáveis.
- Em ambientes com infraestrutura DNS própria, pode haver custo com **HSM (Hardware Security Module)** para armazenamento seguro das chaves criptográficas, representando o principal custo financeiro do controle.

🔧 Custo operacional estimado: Médio

- A configuração inicial exige **profissional qualificado** para geração e publicação correta das chaves (KSK e ZSK).
- Em provedores DNS gerenciados, a ativação pode ser tão simples quanto **habilitar uma opção no painel de controle**, reduzindo significativamente o esforço inicial.

🛡️ Benefício estimado: Alto

- O DNSSEC protege contra ataques que **afetam todos os usuários do domínio simultaneamente**.
- Os ataques mitigados são frequentemente **invisíveis para a vítima**: como o site falso pode ser idêntico ao legítimo, o usuário não tem como perceber que está sendo enganado e não pode se proteger por conta própria.

- Isso reforça o valor do DNSSEC como **controle preventivo aplicado na infraestrutura**, independentemente do comportamento do usuário final.
- O impacto de um ataque bem-sucedido pode incluir **roubo de credenciais em larga escala, danos reputacionais e comprometimento de serviços essenciais ao cidadão**.

1.2 Validade de DNSSEC

Como é verificado?

O teste verifica se a assinatura DNSSEC do registro `SOA` é válida, ou seja, se as chaves criptográficas estão ativas, não expiradas e corretamente configuradas, garantindo integridade de ponta a ponta entre o servidor de nomes autoritativo e o resolvedor do cliente.

Riscos da invalidade

Uma assinatura DNSSEC existente, mas inválida, é potencialmente mais prejudicial do que não ter DNSSEC:

Risco	O que acontece
Proteção nula	Resolvedores que validam DNSSEC rejeitam respostas com assinatura inválida, tornando o domínio inacessível para esses clientes.
Interrupção de serviço	Chaves expiradas ou mal configuradas causam falhas de resolução DNS, tornando o site, o e-mail e outros serviços do domínio inacessíveis.
Mesma vulnerabilidade da ausência	Para resolvedores que não validam DNSSEC, uma assinatura inválida não oferece nenhuma proteção adicional em relação à ausência do controle.

Como resolver?

- Monitorar ativamente a validade das chaves DNSSEC, com alertas automáticos antes do vencimento;
- Realizar a rotação periódica das chaves (ZSK recomendada a cada 30 a 90 dias e KSK anualmente);
- Automatizar o processo de renovação de assinaturas sempre que possível;
- Após qualquer renovação de certificado ou alteração de zona DNS, verificar imediatamente a validade do DNSSEC com ferramentas como o DNSViz.

📚 Referências e documentação oficial

Referência	Descrição	URL
DNSViz	Ferramenta visual de análise e diagnóstico de DNSSEC. Permite verificar a validade das assinaturas e identificar problemas na cadeia de confiança	dnsviz.net
Registro.br — Ferramentas DNS	Ferramenta oficial do registro.br para verificação de configuração e validade de DNSSEC em domínios .br	registro.br/tecnologia/ferramentas
IETF RFC 6781	Guia de boas práticas para operação de DNSSEC, incluindo rotação de chaves e gestão do ciclo de vida	datatracker.ietf.org/doc/html/rfc6781
NIST SP 800-81-3	Diretrizes para implantação segura de DNSSEC, incluindo recomendações sobre validade e rotação de chaves	csrc.nist.gov/pubs/sp/800/81/r3/final

⚠️ **Atenção:** a expiração silenciosa das chaves DNSSEC é uma das causas mais comuns de interrupção de serviço em domínios que implementaram o controle corretamente. O monitoramento ativo da validade das assinaturas é tão importante quanto a própria implementação do DNSSEC. Ferramentas como o DNSViz permitem

verificar visualmente toda a cadeia de confiança e identificar problemas antes que causem interrupção de serviço.

💡 **Dica:** provedores DNS gerenciados com suporte a DNSSEC, como o registro.br, geralmente automatizam a rotação de chaves e a renovação de assinaturas, reduzindo significativamente o risco de expiração. Ao escolher um provedor DNS, verifique se ele oferece esse recurso.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A manutenção da validade do DNSSEC não gera custo financeiro adicional além do já incorrido na implementação do controle 1.1 (Existência de DNSSEC). A renovação de chaves e assinaturas é feita nos mesmos sistemas e infraestrutura já existentes.

🔧 Custo operacional estimado: Baixo

- A manutenção da validade resume-se ao monitoramento da expiração das chaves e à sua renovação periódica, **processo que pode ser amplamente automatizado**.
- O principal risco operacional é a negligência: chaves expiradas por falta de monitoramento podem tornar o domínio inacessível, mas esse risco é **eliminado com alertas automáticos e processos bem definidos de rotação**.

🛡️ Benefício estimado: Alto

- Manter a assinatura DNSSEC válida é a **condição necessária para que todo o investimento realizado no controle 1.1 (Existência de DNSSEC) produza efeito prático**.
- Uma assinatura inválida **anula completamente a proteção do DNSSEC** e ainda pode causar interrupção de serviço para clientes que validam DNSSEC, tornando a manutenção contínua da validade tão crítica quanto a existência do controle.

2. HTTPS, Redirecionamento, Compressão e HSTS

O protocolo HTTPS garante que a comunicação entre o navegador do usuário e o site seja feita por um túnel criptografado, protegendo a autenticidade, a confidencialidade e a integridade dos dados trocados. Os controles a seguir detalham os aspectos fundamentais dessa proteção.

2.1 HTTPS Disponível (HTTP/TLS)

O que é?

O HTTPS é o protocolo HTTP combinado com uma camada de segurança fornecida pelo TLS (Transport Layer Security). O TLS garante três propriedades essenciais na comunicação entre o servidor e o usuário:

- **Autenticidade:** o usuário sabe com quem está se comunicando;
- **Confidencialidade:** os dados não podem ser lidos por terceiros;
- **Integridade:** os dados não podem ser alterados no caminho.

O protocolo SSL, predecessor do TLS, está descontinuado. O padrão atual é o TLS 1.3, especificado na RFC 8446.

Como é verificado?

O teste verifica se o site está acessível via HTTPS.

Riscos da ausência

Risco	Consequência
Dados em texto claro	Qualquer intermediário com acesso à rede — provedor, operador de rede sem fio ou invasor posicionado no caminho — pode ler informações como senhas, dados pessoais e conteúdo de formulários
Adulteração do conteúdo em trânsito	Sem proteção de integridade, um intermediário pode alterar a página entregue ao usuário, inserindo scripts maliciosos, anúncios ou informações falsas atribuídas ao órgão

Risco	Consequência
Alerta de "Não seguro" no navegador	Os navegadores modernos sinalizam explicitamente sites sem HTTPS como não seguros, o que compromete a credibilidade institucional e afasta o cidadão do serviço digital

Como resolver?

Instalar um certificado digital válido e configurar o servidor web para responder via HTTPS.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Ferramenta oficial da Mozilla para geração de configurações TLS seguras para os principais servidores web (Nginx, Apache, IIS, entre outros)	ssl-config.mozilla.org
Let's Encrypt — Documentação	Guia oficial para obtenção e instalação de certificados DV gratuitos, com suporte a renovação automática via Certbot	letsencrypt.org/docs
Certbot	Ferramenta oficial para automação da obtenção e renovação de certificados Let's Encrypt nos principais servidores web e sistemas operacionais	certbot.eff.org
SSL Labs — SSL Test	Ferramenta de análise e diagnóstico da configuração HTTPS de um servidor, com avaliação detalhada do certificado, protocolo e cifras	ssllabs.com/ssltest
RFC 5246	Especificação técnica do protocolo TLS 1.2	datatracker.ietf.org/doc/rfc5246/
RFC 8446	Especificação técnica do protocolo TLS 1.3	datatracker.ietf.org/doc/html/rfc8446

⚠️ **Atenção:** certificados do tipo DV (Domain Validation), como os emitidos pela Let's Encrypt, são gratuitos e tecnicamente adequados para criptografar o tráfego HTTPS. Para órgãos públicos, contudo, certificados DV são considerados insuficientes como solução definitiva: validam apenas o controle sobre o domínio, sem atestar a identidade da organização. Certificados OV (Organization Validation) ou EV (Extended Validation) são mais indicados para sites institucionais, por identificarem formalmente a organização responsável.

É importante ajustar a expectativa quanto ao efeito visível dessa escolha: desde 2019, os principais navegadores **deixaram de exibir qualquer indicação diferenciada** para certificados OV e EV — o antigo nome da organização em destaque na barra de endereços não existe mais. O cidadão comum não distingue um certificado DV de um EV sem abrir os detalhes do certificado. O valor da validação organizacional está, portanto, na **rastreabilidade formal da identidade da organização** e no atendimento a exigências normativas, não em um indicador visual para o usuário. Certificados DV podem ser utilizados como solução temporária enquanto se providencia um certificado mais adequado.

💡 **Dica:** utilize o Mozilla SSL Configuration Generator (ssl-config.mozilla.org) selecionando o perfil "Intermediate" para obter uma configuração equilibrada entre segurança e compatibilidade, adequada para serviços públicos ao cidadão. O perfil "Modern" é mais adequado para APIs e sistemas de comunicação entre servidores.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo a Médio

- Certificados do tipo **DV (Domain Validation)**, como os emitidos pela Let's Encrypt, são gratuitos, mas são considerados insuficientes como solução definitiva para órgãos públicos, pois validam apenas o controle sobre o domínio, sem atestar a identidade da organização.
- Certificados do tipo **OV (Organization Validation) ou EV (Extended Validation)**, mais indicados para sites institucionais e governamentais, podem custar entre algumas centenas e alguns milhares de reais por ano, dependendo da autoridade certificadora e do número de domínios cobertos.

- Certificados **Wildcard e Multi-SAN** tendem a ser mais caros, mas eliminam a necessidade de certificados individuais para cada domínio ou subdomínio.

🔧 Custo operacional estimado: Médio

- A obtenção e configuração inicial do certificado exige **profissional com conhecimento técnico em servidores web e criptografia**, especialmente para certificados OV e EV, cujo processo de validação pela Autoridade Certificadora pode demandar documentação e prazo adicionais.
- A renovação de certificados OV e EV envolve **novo processo de validação pela Autoridade Certificadora**, com maior esforço operacional associado.

🛡️ Benefício estimado: Alto

- O HTTPS é a **base de toda a segurança web**. Sua ausência expõe diretamente todos os usuários do site a interceptações e adulterações, tornando o benefício de sua implementação **proporcional ao volume de acessos e à sensibilidade dos dados trafegados**.

2.2 Redirecionamento para HTTPS

O que é?

Mesmo com HTTPS disponível, um usuário pode acessar o site digitando apenas o domínio (sem "https://") ou clicando em um link antigo, iniciando a conexão de forma insegura via HTTP. O redirecionamento automático corrige esse problema, encaminhando qualquer acesso HTTP para HTTPS.

Como é verificado?

O teste verifica se o servidor redireciona automaticamente conexões HTTP para HTTPS, por meio dos códigos de status HTTP 301 (redirecionamento permanente) ou 302 (redirecionamento temporário).

Riscos da ausência

Risco	Consequência
Conexão insegura involuntária	O usuário trafega dados sem criptografia sem perceber, mesmo que o site ofereça HTTPS
Exposição de credenciais e cookies	Tudo o que for enviado durante a navegação em HTTP — inclusive dados de autenticação e cookies de sessão — trafega em texto claro e pode ser capturado
Permanência de acessos pela versão insegura	Sem o redirecionamento permanente, links antigos, favoritos e resultados de mecanismos de busca continuam levando o usuário à versão HTTP do site indefinidamente

Como resolver?

Configurar o servidor web (porta `80`) para redirecionar todas as requisições HTTP para HTTPS (porta `443`) usando o código 301.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — Redirecting HTTP to HTTPS	Guia oficial da Mozilla sobre configuração de redirecionamento HTTP para HTTPS nos principais servidores web	developer.mozilla.org/en-US/docs/Web/HTTP/Redirections
Mozilla SSL Configuration Generator	Ferramenta para geração de configurações seguras para os principais servidores web, incluindo o bloco de redirecionamento HTTP para HTTPS	ssl-config.mozilla.org
RFC 9110 — HTTP Semantics	Especificação técnica dos códigos de redirecionamento HTTP 301 e 302	datatracker.ietf.org/doc/html/rfc9110#section-15.4

💡 **Dica:** prefira sempre o código 301 (redirecionamento permanente) ao 302 (temporário). O 301 instrui navegadores e mecanismos de busca a atualizarem permanentemente seus registros para a URL HTTPS, evitando que requisições futuras sejam iniciadas via HTTP. O 302 deve ser reservado para situações em que o redirecionamento é genuinamente temporário.

⚠️ **Atenção:** o redirecionamento HTTP para HTTPS não substitui o HSTS (controle 2.4). Um usuário que acessa o site pela primeira vez via HTTP ainda está vulnerável a um ataque MitM antes de receber o redirecionamento. O HSTS elimina essa janela de vulnerabilidade instruindo o navegador a sempre usar HTTPS desde a primeira conexão.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- O redirecionamento é feito por **configuração no servidor web já existente**, sem necessidade de aquisição de qualquer produto, licença ou serviço adicional.

🔧 Custo operacional estimado: Baixo

- Trata-se de uma **simples configuração no servidor web**, geralmente realizada em poucas linhas de código ou por painel de controle, sem custos adicionais de infraestrutura.

🛡️ Benefício estimado: Alto

- Sem esse redirecionamento, a **proteção oferecida pelo HTTPS fica incompleta**, pois qualquer acesso inicial via HTTP permanece vulnerável.
- O benefício é alto porque **complementa de forma decisiva o controle anterior** com custo mínimo.

2.3 Ausência de Compressão HTTP

O que é?

A compressão HTTP reduz o tamanho dos dados transferidos, tornando o carregamento do site mais rápido. No entanto, quando usada em conjunto com criptografia, cria uma vulnerabilidade: invasores podem inferir o conteúdo de dados confidenciais analisando variações no tamanho das respostas comprimidas.

Como é verificado?

O teste simula o comportamento de um navegador: envia uma requisição HTTP ao servidor informando que aceita respostas comprimidas (cabeçalho Accept-Encoding: gzip, deflate) e analisa a resposta recebida. Se o servidor retornar o cabeçalho Content-Encoding indicando que o conteúdo foi comprimido, significa que a compressão HTTP está ativa e o controle é considerado não conforme. Um servidor corretamente configurado ignora a solicitação de compressão e retorna o conteúdo sem comprimir.

Riscos da presença

O risco decorrente da compressão HTTP é o **ataque BREACH**: observando como o tamanho das respostas comprimidas varia conforme o conteúdo enviado, o invasor consegue inferir dados sigilosos presentes na própria resposta, como cookies de sessão e tokens de autenticação, mesmo que a comunicação esteja criptografada.

Como resolver?

Desabilitar a compressão HTTP e TLS no servidor, especialmente para respostas que contenham dados sensíveis.

> ⚠️ **Atenção:** este controle trata da compressão na camada HTTP. A compressão na camada TLS é um problema distinto, abordado no controle 3.6 (Compressão TLS) deste manual. Ambas devem ser desabilitadas, mas por mecanismos diferentes no servidor.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — HTTP compression	Documentação oficial da Mozilla sobre compressão HTTP, seus mecanismos e implicações de segurança	developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Compression
BREACH — Artigo original	Artigo técnico original descrevendo o ataque BREACH e suas condições de exploração	breachattack.com
IETF RFC 7457	Documento que sumariza os ataques conhecidos contra TLS e DTLS, incluindo CRIME e BREACH	datatracker.ietf.org/doc/html/rfc7457
OWASP — Testing for HTTP Verb Tampering	Guia da OWASP sobre testes de segurança relacionados à configuração HTTP, incluindo compressão	owasp.org/www-project-web-security-testing-guide/

> 💡 **Dica:** a desabilitação da compressão HTTP pode causar um leve aumento no volume de dados transferidos. Na prática, esse impacto é irrelevante para a maioria dos sites institucionais, cujo conteúdo principal são páginas HTML, documentos e formulários — tipos de conteúdo em que o ganho de compressão é modesto. O impacto seria mais perceptível em sites que transferem grandes volumes de dados textuais em alta frequência.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Desabilitar a compressão é uma **alteração de configuração no servidor web já existente**, sem custo financeiro direto.
- O único impacto indireto é um possível **leve aumento no consumo de banda**, que em contratos com cobrança por tráfego pode representar um custo marginal, geralmente irrelevante.

🔧 Custo operacional estimado: Baixo

- A desabilitação da compressão resume-se a **poucas linhas de configuração no servidor web**.
- Não exige conhecimento técnico avançado, sendo **uma das alterações de configuração mais simples** abordadas neste manual.
- O único impacto operacional indireto é um possível leve aumento no volume de dados transferidos, o que pode exigir **monitoramento pontual em ambientes com links de capacidade muito limitada** ou contratos com cobrança por tráfego.

🛡️ Benefício estimado: Médio

- O controle mitiga ataques da classe **CRIME e BREACH**, que exploram a interação entre compressão e criptografia para inferir o conteúdo de dados sensíveis, como cookies de sessão e tokens de autenticação.
- O benefício é classificado como médio porque a exploração exige **condições específicas**: o atacante precisa estar em posição de observar o tráfego criptografado e, em alguns casos, ser capaz de induzir o servidor a incluir conteúdo controlado pelo atacante nas respostas comprimidas.
- A **facilidade e o custo nulo** de mitigação tornam o controle amplamente justificável: o esforço de implementação é mínimo em relação ao risco eliminado.
- Sites que transmitem **dados sensíveis de cidadãos**, como sistemas de autenticação, formulários e consultas a serviços públicos, têm benefício proporcionalmente maior.

2.4 HSTS (HTTP Strict Transport Security)

O que é?

O HSTS é um cabeçalho de segurança que determina ao navegador a utilização exclusiva do protocolo HTTPS em acessos subsequentes ao domínio, mesmo mediante a inserção manual do prefixo 'http://' pelo usuário. Essa diretriz bloqueia preventivamente qualquer conexão HTTP, suprimindo a janela de vulnerabilidade existente antes do redirecionamento.

Como é verificado?

O teste verifica se o cabeçalho HSTS está presente já no primeiro contato com o servidor, com um tempo de validade de pelo menos um ano (`max-age=31536000`).

Riscos da ausência

Risco	Consequência
Ataque de downgrade (SSL Stripping)	O invasor impede o redirecionamento para HTTPS, mantendo a conexão insegura sem que o usuário perceba
Roubo de cookies de sessão	Cookies transmitidos na conexão HTTP inicial podem ser capturados antes do redirecionamento

Como resolver?

O HSTS deve ser implantado de forma gradual, em três etapas. Ativar imediatamente o HSTS com validade longa e cobertura de subdomínios sem validação prévia pode tornar o site inacessível para usuários caso haja algum problema de configuração HTTPS ainda não identificado.

Etapa	Configuração	Duração mínima antes de avançar
1. Teste	`max-age=300` (5 minutos)	1 semana sem problemas
2. Consolidação	`max-age=31536000` (1 ano)	1 mês sem problemas
3. Definitiva	`max-age=63072000; includeSubDomains; preload` (2 anos)	Permanente

⚠️ **Atenção:** antes de avançar para a etapa 3, confirmar que o HTTPS funciona corretamente no domínio principal e em todos os subdomínios. A inclusão na lista

de preload é de difícil reversão e pode levar meses para ser desfeita, pois depende de atualização dos navegadores.

Com HSTS ativo, falhas relacionadas ao HTTPS — como certificado expirado, cadeia inválida ou domínio incompatível — podem impedir o acesso pelo navegador de forma mais rígida, sem possibilidade de exceção pelo usuário. Isso reforça a necessidade de monitoramento automatizado da validade dos certificados e de validação prévia do domínio principal e dos subdomínios antes da ativação com `includeSubDomains` ou `preload`.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — HTTP Strict Transport Security	Documentação oficial da Mozilla sobre o cabeçalho HSTS, seus parâmetros e instruções de implementação	developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Headers/Strict-Transport-Security
HSTS Preload List	Ferramenta oficial para submissão de domínios à lista de pré-carregamento dos navegadores, com verificação de elegibilidade	hstspreload.org
Mozilla SSL Configuration Generator	Gera automaticamente a configuração HSTS correta para os principais servidores web	ssl-config.mozilla.org
RFC 6797	Especificação técnica do protocolo HSTS	datatracker.ietf.org/doc/html/rfc6797
OWASP — HTTP Strict Transport Security Cheat Sheet	Guia prático da OWASP sobre implementação e boas práticas do HSTS	cheatsheetseries.owasp.org/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html

💡 **Dica:** utilize o SSL Labs (ssllabs.com/ssltest) para verificar se o cabeçalho HSTS está sendo enviado corretamente e se os parâmetros estão de acordo com as boas

práticas. O relatório indica explicitamente se o domínio está elegível para a lista de preload e quais requisitos ainda precisam ser atendidos.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Assim como o redirecionamento, o HSTS é implementado pela **adição de um cabeçalho HTTP na configuração do servidor web existente**, sem necessidade de aquisição de produtos ou serviços adicionais.

🔧 Custo operacional estimado: Médio

- A configuração em si resume-se à **adição de uma linha na configuração do servidor web**, o que é tecnicamente simples.
- O custo operacional real está na **implantação segura**: é necessário confirmar que o HTTPS funciona corretamente no domínio principal e em todos os subdomínios antes de ativar o HSTS com cobertura ampla.
- A implantação deve ser feita em **etapas graduais ao longo de semanas**, não de forma imediata.
- Erros de configuração podem **tornar o site inacessível pelo período de validade configurado**, o que em domínios governamentais pode ter impacto direto no cidadão.
- A decisão de submeter o domínio à **lista de pré-carregamento dos navegadores (preload)** é de difícil reversão e exige análise cuidadosa antes de ser tomada.

🛡️ Benefício estimado: Alto

- O HSTS elimina uma **vulnerabilidade residual que persiste mesmo quando HTTPS e redirecionamento estão configurados corretamente**.
- Ataques de **SSL Stripping** são silenciosos e difíceis de detectar pelo usuário, tornando essa proteção adicional de alto valor com custo praticamente nulo.

3. TLS

3.1 Versão do TLS

O que é?

O TLS (Transport Layer Security) é o protocolo que criptografa a comunicação entre o navegador e o servidor. Com o tempo, versões mais antigas do TLS foram descontinuadas por conterem vulnerabilidades conhecidas e publicamente exploráveis. A versão atual e recomendada é o TLS 1.3, definida na RFC 8446.

> ⚠️ Os principais navegadores já removeram o suporte a TLS 1.0 e 1.1. Sites que não oferecem pelo menos TLS 1.2 ficam inacessíveis para esses usuários.

Como é verificado?

O teste verifica se o servidor web oferece suporte exclusivamente a versões seguras do TLS (1.2 e/ou 1.3), sem aceitar conexões por versões obsoletas.

Riscos do não atendimento

Versões antigas do TLS possuem vulnerabilidades documentadas que permitem que invasores quebrem ou contornem a criptografia:

Risco	Consequência
Downgrade	O invasor força o uso de uma versão mais fraca do TLS para explorar suas vulnerabilidades conhecidas
POODLE	O invasor provoca falhas nas versões modernas para forçar o uso do SSL 3.0, protocolo com falha grave de segurança
BEAST	O invasor realiza ataques MitM para obter ilegalmente informações de uma sessão ativa

Risco	Consequência
Bleichenbacher / ROBOT	O invasor explora falhas na forma como o servidor trata o preenchimento (padding) das mensagens RSA e, a partir das respostas de erro obtidas, consegue decifrar a comunicação — ataque conhecido desde 1998 e redescoberto em 2017 sob o nome ROBOT

Como resolver?

Configurar o servidor web para aceitar apenas TLS 1.2 e TLS 1.3, desativando explicitamente todas as versões anteriores (SSL 2.0, SSL 3.0, TLS 1.0 e TLS 1.1).

> 💡 **Nota:** para serviços públicos ao cidadão, recomenda-se manter o suporte ao TLS 1.2 por compatibilidade com dispositivos mais antigos. O TLS 1.3 exclusivo é mais adequado para APIs e sistemas de comunicação entre servidores, onde o controle sobre os clientes é total. Consulte a discussão sobre perfis de compatibilidade na seção 3.2 deste manual.

> ⚠️ **Atenção:** a desativação do TLS 1.0 e 1.1 pode interromper a comunicação com sistemas legados, como sistemas de arrecadação, portais antigos ou integrações com terceiros. Antes da aplicação definitiva da mudança, recomenda-se que a equipe de TI mapeie o tráfego atual, identifique eventuais dependências legadas e valide as alterações em ambiente de homologação, a fim de garantir a continuidade dos serviços.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Ferramenta para geração de configurações TLS seguras para os principais servidores web, com seleção de perfil de compatibilidade (Modern, Intermediate, Old)	ssl-config.mozilla.org
SSL Labs — SSL Test	Ferramenta de análise da configuração TLS do servidor, com identificação das versões suportadas e avaliação de segurança	ssllabs.com/ssltest

Referência	Descrição	URL
RFC 8446	Especificação técnica do TLS 1.3	datatracker.ietf.org/doc/html/rfc8446
RFC 8996	Documento que deprecia formalmente o TLS 1.0 e TLS 1.1	datatracker.ietf.org/doc/html/rfc8996
NIST SP 800-52 Rev. 2	Diretrizes do NIST para seleção, configuração e uso de protocolos TLS, incluindo versões recomendadas e perfis de configuração	csrc.nist.gov/pubs/sp/800/52/r2/final

💡 **Dica:** utilize o SSL Labs (ssllabs.com/ssltest) para verificar quais versões do TLS seu servidor aceita atualmente. O relatório identifica claramente versões obsoletas ainda ativas e apresenta uma avaliação geral de segurança da configuração TLS do servidor.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A desativação de versões antigas do TLS é feita por configuração no servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Médio

- O custo operacional médio é justificado por três fatores: compatibilidade — sistemas ou clientes legados que dependem de versões antigas do TLS podem deixar de funcionar após a mudança, exigindo mapeamento prévio dos impactos.
- Testes — é necessário validar que o servidor continua acessível e funcional após a desativação das versões antigas.
- Manutenção contínua — à medida que novas vulnerabilidades surgem, as configurações precisam ser revisadas periodicamente para garantir que apenas versões seguras permaneçam habilitadas.

🛡️ Benefício estimado: Médio

- O benefício é médio porque: a exploração das vulnerabilidades das versões antigas exige condições específicas, como a capacidade do invasor de interceptar o tráfego e forçar um downgrade de protocolo.
- No entanto, as vulnerabilidades são públicas e bem documentadas, com ferramentas de exploração amplamente disponíveis, o que eleva o risco em ambientes expostos à internet.
- A combinação de custo financeiro nulo com benefício médio torna o controle de relação custo-benefício claramente favorável à implementação.

3.2 Cifras (Seleção de Algoritmos)

O que é?

Durante o estabelecimento de uma conexão TLS (processo chamado de handshake), o servidor e o cliente negociam quais algoritmos criptográficos, chamados de cifras, usarão para proteger a comunicação. Com o tempo, vulnerabilidades foram descobertas em diversas cifras, tornando-as inseguras.

> 💡 **Referência rápida:** a especificação base do TLS 1.2 (RFC 5246) define 37 conjuntos de cifras, e o registro oficial da IANA acumula centenas de combinações utilizáveis com essa versão — nem todas seguras. O TLS 1.3 define apenas 5 conjuntos, e todos são considerados seguros.

Cada conjunto de cifras combina algoritmos com funções distintas. São exemplos de componentes obsoletos que, apesar de ainda amplamente encontrados, não devem mais ser utilizados: RC4 (cifragem), DSA (assinatura), MD5 (função hash) e DH e ECDH em suas variantes estáticas, sem sigilo futuro (troca de chaves).

Como é verificado?

O teste verifica se o servidor web aceita conexões apenas com cifras consideradas seguras, conforme as diretrizes do NIST SP 800-52 Rev. 2 e da OWASP.

Riscos do não atendimento

O uso de cifras obsoletas expõe o servidor a ataques conhecidos e documentados:

Risco	Consequência
Bar Mitzvah	Explora a geração fraca de chaves do algoritmo RC4 para obter os primeiros bytes de uma sessão TLS
FREAK	Força o servidor a aceitar criptografia obsoleta via downgrade, permitindo a quebra das chaves
LOGJAM	Força o servidor a usar parâmetros de troca de chaves fracos, viabilizando a interceptação da comunicação
RACCOON	Explora a troca de chaves Diffie-Hellman no TLS 1.2 e versões anteriores para descriptografar a conexão
SWEET32	Explora cifras de bloco de 64 bits, como o 3DES, em sessões longas ou com grande volume de tráfego, aumentando a chance de colisões criptográficas que podem permitir a recuperação parcial de informações sensíveis

Como resolver?

Configurar o servidor para:

- Suportar apenas cifras modernas, conforme lista do NIST SP 800-52 Rev. 2;
- Impor a preferência de ordem do servidor sobre a do cliente, impedindo que o cliente force o uso de uma cifra mais fraca.

💡 **Nota sobre compatibilidade:** para serviços públicos ao cidadão, recomenda-se o perfil Intermediate do Mozilla SSL Configuration Generator, que oferece o equilíbrio adequado entre segurança e compatibilidade. O perfil Modern é mais adequado para APIs e sistemas de comunicação entre servidores. Em qualquer caso, cifras obsoletas como RC4, 3DES, MD5 e EXPORT nunca devem ser habilitadas, independentemente de alegações de compatibilidade.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Ferramenta para geração de listas de cifras seguras e ordenadas para os principais servidores web, com três perfis de compatibilidade (Modern, Intermediate, Old)	ssl-config.mozilla.org
Mozilla — Cipher Suite Recommen-dations	Documentação da Mozilla sobre cifras recomendadas e obsoletas, com justificativas técnicas para cada decisão	wiki.mozilla.org/Security/Server_Side_TLS
NIST SP 800-52 Rev. 2	Diretrizes do NIST para seleção de cifras TLS, incluindo listas de algoritmos aprovados e proibidos	csrc.nist.gov/pubs/sp/800/52/r2/final
OWASP — TLS Cheat Sheet	Guia prático da OWASP sobre configuração segura de TLS, incluindo seleção de cifras	cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html
SSL Labs — SSL Test	Ferramenta de análise das cifras aceitas pelo servidor, com identificação de algoritmos obsoletos e avaliação de segurança	ssllabs.com/ssltest

💡 **Dica:** o Mozilla SSL Configuration Generator (ssl-config.mozilla.org) é a forma mais prática e confiável de obter uma lista de cifras seguras e atualizada para o servidor específico em uso. Basta selecionar o servidor web, a versão instalada e o perfil de compatibilidade desejado. A ferramenta gera automaticamente a configuração completa, incluindo a ordem correta das cifras.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A seleção de cifras é feita por configuração no servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo a Médio

- A configuração resume-se à edição de parâmetros no servidor web, com base em listas de cifras recomendadas já disponíveis publicamente (ex.: Mozilla SSL Configuration Generator).
- O principal cuidado é verificar se sistemas ou clientes legados dependem de cifras obsoletas, o que pode exigir um mapeamento prévio antes da mudança.
- Ferramentas gratuitas como o SSL Labs permitem validar rapidamente se as cifras configuradas são adequadas.

🛡️ Benefício estimado: Médio

- A exploração de cifras obsoletas geralmente exige que o invasor já esteja em posição de interceptar o tráfego, o que não é trivial.
- No entanto, as vulnerabilidades são públicas e amplamente documentadas, com ferramentas de exploração disponíveis, elevando o risco em ambientes expostos à internet.
- A combinação de custo praticamente nulo com proteção contra uma classe conhecida de ataques torna o controle claramente vantajoso.

3.3 Ordem das Cifras

O que é?

Quando um navegador se conecta a um servidor, os dois negociam qual cifra usarão para proteger a comunicação. Nessa negociação, o navegador envia uma lista de cifras que

suporta, geralmente ordenada da mais forte para a mais fraca. O servidor, por sua vez, pode:

- Respeitar a ordem do navegador, usando a primeira cifra da lista que também suporta, ou
- Impor a própria ordem de preferência, escolhendo a cifra que considera mais adequada, independentemente da preferência do navegador

> ⚠️ Sem que o servidor imponha sua própria ordem, um cliente mal-intencionado pode manipular a lista enviada para forçar o uso da cifra mais fraca disponível, mesmo que cifras mais fortes estejam disponíveis, caracterizando um ataque de downgrade passivo.

Como é verificado?

O teste verifica dois aspectos:

- **Imposição da preferência do servidor:** o servidor ignora a ordem proposta pelo cliente e usa a própria ordem de preferência;
- **Ordem recomendada:** as cifras são oferecidas em ordem decrescente de segurança, priorizando as mais fortes sobre as suficientes, e estas sobre as desatualizadas.

Riscos do não atendimento

Risco	Consequência
Downgrade passivo	Um cliente mal-intencionado manipula a lista de cifras para forçar o uso da cifra mais fraca aceita pelo servidor
Exploração de cifras fracas	Sem ordem definida pelo servidor, a negociação pode resultar no uso de cifras com vulnerabilidades conhecidas, expondo a conexão aos mesmos ataques listados no controle 3.2 (FREAK, LOGJAM, RACCOON, entre outros)

Como resolver?

Configurar o servidor web para:

- Impor sua própria ordem de preferência de cifras, ignorando a ordem sugerida pelo cliente;
- Ordenar as cifras do mais forte para o mais fraco, garantindo que a negociação sempre resulte na cifra mais segura disponível entre as duas partes.

> 💡 **Serviços públicos ao cidadão:** recomenda-se o perfil intermediate do Mozilla SSL Configuration Generator, que oferece o equilíbrio adequado entre segurança e compatibilidade, cobrindo mais de 99% dos navegadores ativos. O endurecimento máximo (perfil modern, com TLS 1.3 exclusivo) é mais adequado para APIs e sistemas de comunicação entre servidores, onde o controle sobre os clientes é total.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Ferramenta para geração de listas de cifras seguras e ordenadas para os principais servidores web, com três perfis de compatibilidade (Modern, Intermediate, Old)	ssl-config.mozilla.org
Mozilla — Cipher Suite Recommendations	Documentação da Mozilla sobre cifras recomendadas e obsoletas, com justificativas técnicas para cada decisão	wiki.mozilla.org/Security/Server_Side_TLS
NIST SP 800-52 Rev. 2	Diretrizes do NIST para seleção de cifras TLS, incluindo listas de algoritmos aprovados e proibidos	csrc.nist.gov/pubs/sp/800/52/r2/final
OWASP — TLS Cheat Sheet	Guia prático da OWASP sobre configuração segura de TLS, incluindo seleção de cifras	cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

Referência	Descrição	URL
SSL Labs — SSL Test	Ferramenta de análise das cifras aceitas pelo servidor, com identificação de algoritmos obsoletos e avaliação de segurança	ssllabs.com/ssltest

💡 **Dica:** o Mozilla SSL Configuration Generator (ssl-config.mozilla.org) é a forma mais prática de obter a configuração correta de ordem de cifras. A ferramenta gera automaticamente tanto a lista de cifras quanto a diretiva de preferência do servidor, eliminando a necessidade de definição manual da ordem.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A configuração da ordem de cifras é feita nos arquivos de configuração do servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo

- A configuração é simples e bem documentada para os principais servidores web.
- Ferramentas gratuitas como o Mozilla SSL Configuration Generator e o SSL Labs eliminam grande parte do esforço técnico, gerando e validando configurações prontas.
- O principal cuidado é garantir que a ordem definida não quebre a compatibilidade com clientes legítimos que suportam apenas cifras mais antigas.

🛡️ Benefício estimado: Médio

- Este controle complementa o controle 3.2 (seleção de cifras), não sendo eficaz de forma isolada: de nada adianta ordenar corretamente cifras inseguras.
- No entanto, quando combinado com a seleção adequada de cifras, elimina o risco de downgrade passivo, garantindo que a negociação sempre resulte na proteção mais forte possível.

- A facilidade de implementação e o custo nulo tornam a relação custo-benefício amplamente favorável.

3.4 Parâmetros de Troca de Chaves

O que é?

Antes de iniciar uma comunicação criptografada, o servidor e o cliente precisam combinar uma chave secreta compartilhada, sem que essa chave trafegue pela rede. O método mais utilizado para isso é o Diffie-Hellman (DH), que permite duas partes chegarem a um segredo comum mesmo em um canal inseguro.

Existem três variantes principais desse método:

Método	Descrição	Status
ECDHE	Diffie-Hellman Efêmero com curvas elípticas, mínimo de 224 bits	✅ **Recomendado**
DHE	Diffie-Hellman Efêmero com grupos de campos finitos, mínimo de 2048 bits	✅ **Recomendado**
RSA	Alternativa para troca de chaves, sem a propriedade de sigilo futuro	⚠️ **Desatualizado para troca de chaves**

💡 **Por que "efêmero" importa?** Métodos efêmeros (ECDHE e DHE) geram uma chave nova a cada sessão. Isso garante o sigilo futuro (Forward Secrecy): mesmo que a chave privada do servidor seja comprometida no futuro, sessões passadas permanecem protegidas, pois suas chaves já foram descartadas.

Como é verificado?

O teste verifica se os parâmetros usados na troca de chaves são suficientemente seguros:

- **ECDHE:** verifica se as curvas elípticas utilizadas têm pelo menos 224 bits, conforme recomendado pelo NIST SP 800-186 e pela RFC 7748
- **DHE:** verifica se os grupos de campos finitos utilizados seguem os grupos pré-definidos e seguros da RFC 7919

Riscos do não atendimento

O uso de parâmetros fracos ou inadequados na troca de chaves expõe a conexão a:

Risco	Consequência
Ataque de timing (side-channel)	O invasor mede o tempo de processamento das operações criptográficas para inferir informações sobre a chave privada
Ataque de cache (side-channel)	O invasor monitora padrões de acesso à memória cache do processador para extrair informações sobre a chave
Quebra de chaves fracas	Parâmetros DH com tamanho insuficiente podem ser quebrados com poder computacional acessível, comprometendo retroativamente sessões gravadas

⚠️ Curvas elípticas inadequadas são especialmente suscetíveis a ataques side-channel, nos quais o invasor não ataca a matemática da criptografia diretamente, mas sim a forma como ela é executada no hardware.

Como resolver?

- Usar ECDHE com curvas recomendadas pelo NIST (ex.: P-256, P-384) ou pela RFC 7748 (ex.: X25519, X448), que oferecem resistência nativa a ataques side-channel
- Usar DHE com grupos de campos finitos de pelo menos 2048 bits, conforme a RFC 7919
- Evitar RSA como método de troca de chaves, reservando seu uso apenas para verificação de certificados

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente a configuração de curvas elípticas e parâmetros DHE seguros para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre parâmetros de troca de chaves recomendados, incluindo curvas elípticas e grupos DHE	wiki.mozilla.org/Security/Server_Side_TLS
NIST SP 800-186	Recomendações do NIST para uso de curvas elípticas em criptografia, incluindo lista de curvas aprovadas	csrc.nist.gov/publications/detail/sp/800-186/final
RFC 7748	Especificação das curvas elípticas X25519 e X448, recomendadas por sua resistência a ataques side-channel	datatracker.ietf.org/doc/html/rfc7748
RFC 7919	Especificação dos grupos Diffie-Hellman pré-definidos e seguros para uso no TLS	datatracker.ietf.org/doc/html/rfc7919
SSL Labs — SSL Test	Verifica os parâmetros de troca de chaves utilizados pelo servidor, incluindo curvas elípticas e tamanho dos grupos DHE	ssllabs.com/ssltest

💡 **Dica:** o Mozilla SSL Configuration Generator (ssl-config.mozilla.org) configura automaticamente as curvas elípticas e os parâmetros DHE recomendados para cada perfil de compatibilidade, eliminando a necessidade de seleção manual. Para ambientes que exigem conformidade com o NIST, selecione o perfil Intermediate ou Modern e verifique as curvas geradas em relação à lista aprovada no NIST SP 800-186.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A configuração dos parâmetros de troca de chaves é feita nos arquivos de configuração do servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à seleção das curvas elípticas e grupos DH nos arquivos do servidor web, com base em listas recomendadas e bem documentadas.
- Ferramentas como o Mozilla SSL Configuration Generator e o SSL Labs facilitam tanto a geração quanto a validação das configurações.
- O principal ponto de atenção é garantir compatibilidade com clientes legítimos, especialmente sistemas mais antigos que podem não suportar as curvas mais modernas.

🛡️ Benefício estimado: Médio

- Ataques side-channel exigem acesso privilegiado ao ambiente de execução do servidor, o que não é trivial em infraestruturas bem configuradas.
- No entanto, o uso de parâmetros fracos pode comprometer retroativamente sessões gravadas por invasores, especialmente quando combinado com a ausência de sigilo futuro (Forward Secrecy).
- A adoção de curvas modernas como X25519 é uma tendência consolidada e progressivamente obrigatória, tornando o controle relevante também para conformidade futura.

3.5 Função Hash para Troca de Chaves

O que é?

Durante o handshake TLS, o servidor prova sua identidade ao cliente por meio de uma assinatura digital, calculada com base em todas as mensagens trocadas até aquele momento. Essa assinatura é gerada com uma função hash, que transforma os dados em um valor único e fixo, funcionando como uma "impressão digital" da comunicação.

Se a função hash utilizada for fraca, essa impressão digital pode ser falsificada, comprometendo a autenticidade do servidor.

> 💡 **Analogia:** a função hash atua como o lacre de segurança de um envelope. Se o lacre for frágil e fácil de falsificar, um atacante pode violar o envelope, alterar seu conteúdo e relacrá-lo sem que o destinatário perceba a violação.

As funções hash recomendadas atualmente são SHA-2 e SHA-3. O SHA-1 e versões anteriores estão descontinuados e não devem mais ser utilizados.

Como é verificado?

O teste verifica se o servidor web utiliza exclusivamente funções hash seguras, como SHA-2 ou SHA-3, para gerar as assinaturas digitais durante a troca de chaves.

Riscos do não atendimento

O uso de funções hash obsoletas, como SHA-1 e MD5, expõe a comunicação a ataques bem documentados:

Risco	Consequência
Colisão de Hash	O atacante cria duas mensagens diferentes que resultam na mesma "impressão digital" (valor de hash). Essa falha no algoritmo permite, por exemplo, falsificar certificados digitais, fazendo com que sistemas de segurança os aceitem como legítimos. O ataque SHAttered demonstrou isso na prática contra o SHA-1 em 2017

Risco	Consequência
Ataque de Pré-imagem	O invasor conhece apenas o hash de uma mensagem e tenta reconstituir qualquer mensagem que o gere, sem conhecer a mensagem original. Compromete a confidencialidade dos dados
Ataque de Segunda Pré-imagem	O invasor parte de uma mensagem legítima já existente e busca criar uma segunda mensagem diferente que produza o mesmo hash. É mais perigoso na prática, pois permite substituir um documento ou comunicação real por um falsificado de forma indetectável

⚠️ O ataque SHAttered (2017) demonstrou que colisões no SHA-1 são computacionalmente viáveis. O NIST determinou a transição obrigatória para SHA-2 ou SHA-3 em todas as aplicações federais americanas.

Como resolver?

Configurar o servidor web para utilizar exclusivamente SHA-2 (ex.: SHA-256, SHA-384) ou SHA-3 nas assinaturas digitais durante o handshake TLS, desabilitando explicitamente o uso de SHA-1 e MD5.

💡 **Nota:** O TLS 1.3 eliminou o suporte a SHA-1 e MD5 por design (RFC 8446). Servidores que operam com TLS 1.3 exclusivo estão protegidos independentemente das configurações de cifras. As configurações acima são especialmente relevantes para garantir segurança no TLS 1.2, mantido por compatibilidade.

Resumo das funções hash

Função Hash	Status	Utilização nas cifras configuradas
SHA-256	✅ **Recomendada**	*-AES128-GCM-SHA256, *-CHACHA20-POLY1305
SHA-384	✅ **Recomendada**	*-AES256-GCM-SHA384

Função Hash	Status	Utilização nas cifras configuradas
SHA-512	✅ Recomendada	Disponível mediante configuração explícita dos algoritmos de assinatura no servidor web
SHA-1	❌ Descontinuada	Ausente em todas as cifras listadas
MD5	❌ Quebrada	Ausente em todas as cifras listadas

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que excluem SHA-1 e MD5 das assinaturas digitais, para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre algoritmos de assinatura recomendados e obsoletos no contexto do TLS	wiki.mozilla.org/Security/Server_Side_TLS
NIST FIPS 180-4	Especificação oficial das funções hash SHA-2 aprovadas pelo NIST	csrc.nist.gov/publications/detail/fips/180/4/final
NIST SP 800-131A Rev. 2	Diretrizes do NIST para transição de algoritmos criptográficos, incluindo a descontinuação formal do SHA-1	csrc.nist.gov/pubs/sp/800/131/a/r2/final
SHAttered — Ataque ao SHA-1	Documentação técnica do ataque SHAttered, que demonstrou a viabilidade prática de colisões no SHA-1	marc-stevens.nl/research/shattered.io/
SSL Labs — SSL Test	Verifica os algoritmos de assinatura utilizados pelo servidor durante o handshake TLS, identificando uso de SHA-1 ou MD5	ssllabs.com/ssltest
RFC 8446	Especificação do TLS 1.3, que eliminou o suporte a SHA-1 e MD5 por design	datatracker.ietf.org/doc/html/rfc8446

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- A configuração das funções hash é feita nos arquivos de configuração do servidor web já existente. Não há necessidade de aquisição de produtos, licenças ou serviços adicionais.
- **Atenção:** caso o certificado digital atual utilize SHA-1 em sua assinatura, será necessário renová-lo com SHA-2, o que pode envolver custo de emissão dependendo da autoridade certificadora. Atualmente, porém, a emissão de certificados com SHA-1 já é bloqueada pelos principais navegadores, tornando esse cenário pouco provável em infraestruturas recentes.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à remoção de funções hash obsoletas nos parâmetros do servidor web, com base em documentação amplamente disponível.
- Ferramentas gratuitas como SSL Labs permitem verificar rapidamente quais funções hash estão em uso e se estão em conformidade com as recomendações.
- O principal ponto de atenção é verificar a compatibilidade com clientes legados que possam depender de SHA-1, situação cada vez mais rara dado que os principais navegadores já bloqueiam conexões com esse algoritmo.

🛡️ Benefício estimado: Médio

- A exploração prática de colisões SHA-1, embora demonstrada, ainda exige poder computacional significativo, limitando o número de invasores capazes de realizá-la.
- No entanto, a tendência é que esse custo computacional diminua progressivamente, tornando o risco crescente ao longo do tempo.
- A substituição por SHA-2 ou SHA-3 é uma medida simples, consolidada e sem custo financeiro, tornando a relação custo-benefício amplamente favorável à implementação imediata.

3.6 Compressão TLS

O que é?

A compressão TLS é um recurso que cria uma vulnerabilidade ao reduzir o tamanho dos dados antes de criptografá-los. O objetivo é a economia de banda. No entanto, essa combinação de compressão e criptografia permite que o invasor tenha acesso a informações sobre o conteúdo dos dados através do seu tamanho, mesmo que estejam criptografados.

> 💡 **Por que isso é um problema?** A compressão funciona eliminando repetições nos dados. Quanto mais repetições, menor o resultado. Um invasor pode explorar isso injetando trechos conhecidos na comunicação e observando se o tamanho da resposta diminui, o que confirmaria que aquele trecho já estava presente nos dados secretos.

> ⚠️ **Importante:** este controle trata da compressão na camada TLS, distinta da compressão HTTP abordada no controle 2.3 (Ausência de Compressão HTTP). Ambas devem ser desativadas, mas por mecanismos diferentes no servidor.

Como é verificado?

O teste verifica se o servidor web oferece suporte à compressão na camada TLS. O resultado esperado é que esse suporte não esteja disponível.

Riscos da presença da compressão TLS

Risco	Consequência
CRIME (Compression Ratio Info-leak Made Easy)	O invasor injeta trechos de texto na comunicação e observa variações no tamanho das respostas comprimidas para inferir o conteúdo de cookies de sessão, podendo sequestrar a sessão do usuário
Compression-based Side Channel	De forma similar ao CRIME, o invasor usa variações de tamanho causadas pela compressão para reconstruir partes secretas da comunicação, realizando um grande número de requisições controladas

⚠️ O ataque CRIME foi demonstrado publicamente em 2012 e levou os principais navegadores e servidores a desativarem a compressão TLS. O TLS 1.3 removeu o suporte à compressão por completo, eliminando essa classe de ataques nas conexões mais modernas.

Como resolver?

Desativar explicitamente o suporte à compressão na camada TLS no servidor web. Em servidores que utilizam TLS 1.3 exclusivamente, essa configuração já é desnecessária, pois o protocolo não oferece mais esse recurso.

Comparativo: compressão TLS × compressão HTTP

Característica	Compressão TLS	Compressão HTTP
Onde ocorre a compressão	Protocolo TLS	Protocolo HTTP
Ataque relacionado	CRIME (CVE-2012-4929)	BREACH (CVE-2013-3587)
TLS 1.3	Removida por design (RFC 8446)	Não afetada pelo protocolo TLS
Seção deste manual	3.6	2.3

⚠️ **Atenção:** desabilitar a compressão TLS não desabilita a compressão HTTP e vice-versa. Ambas devem ser desabilitadas separadamente para mitigação completa. Consulte o controle 2.3 (Ausência de Compressão HTTP) deste manual para a configuração da compressão HTTP.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que desabilitam a compressão TLS para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre compressão TLS, com explicação do ataque CRIME e recomendações de configuração	wiki.mozilla.org/Security/Server_Side_TLS
IETF RFC 7457	Documento que sumariza os ataques conhecidos contra TLS e DTLS, incluindo o CRIME e ataques baseados em compressão	datatracker.ietf.org/doc/html/rfc7457
RFC 8446	Especificação do TLS 1.3, que removeu o suporte à compressão por design	datatracker.ietf.org/doc/html/rfc8446
SSL Labs — SSL Test	Verifica se o servidor oferece suporte à compressão TLS, identificando o item "SSL/TLS compression" no relatório	ssllabs.com/ssltest

> 💡 **Dica:** utilize o SSL Labs (ssllabs.com/ssltest) para verificar rapidamente se a compressão TLS está desabilitada no servidor. O relatório indica explicitamente o item "SSL/TLS compression", que deve exibir o valor "No". O Mozilla SSL Configuration Generator garante que a configuração gerada desabilita a compressão TLS por padrão em todos os perfis disponíveis.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A desativação da compressão TLS é feita por configuração no servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à desativação de um parâmetro no servidor web, bem documentada para os principais servidores (Apache, Nginx, IIS).
- Servidores que já utilizam TLS 1.3 exclusivamente não precisam de nenhuma ação adicional, pois o protocolo não suporta compressão.
- O impacto na performance é negligenciável: o leve aumento no consumo de banda causado pela desativação da compressão é compensado pela eliminação do risco.

🛡️ Benefício estimado: Médio

- O ataque CRIME exige que o invasor seja capaz de interceptar o tráfego e injetar requisições na sessão da vítima, o que demanda um posicionamento privilegiado na rede.
- No entanto, o ataque é bem documentado e possui ferramentas públicas de exploração, tornando-o acessível a invasores com conhecimento intermediário.
- Servidores que já adotaram TLS 1.3 exclusivamente já estão protegidos por padrão, reduzindo o universo de sistemas efetivamente vulneráveis.

3.7 Renegociação Segura

O que é?

O TLS permite que, durante uma conexão já estabelecida, o servidor ou o cliente solicite um novo handshake para atualizar os parâmetros criptográficos da sessão, processo chamado de renegociação. Isso pode ser útil, por exemplo, quando o servidor precisa solicitar um certificado do cliente em meio a uma sessão já iniciada.

O problema é que, nas versões TLS 1.0 a 1.2, não havia vínculo criptográfico entre o handshake original e o renegociado. Isso criava uma brecha: um invasor posicionado entre o cliente e o servidor poderia interceptar a conexão, injetar seu próprio tráfego e, em seguida, passar a conexão para o servidor como se fosse legítima.

💡 **Analogia:** imagine que você está no meio de uma ligação telefônica autenticada e, de repente, um terceiro assume a linha sem que nenhum dos lados perceba a troca. O novo interlocutor consegue continuar a conversa como se fosse você.

Esse problema foi corrigido pela RFC 5746, que introduziu a extensão de indicação de renegociação segura (Renegotiation Indication Extension), criando um vínculo criptográfico entre os dois handshakes. A versão anterior, sem esse vínculo, passou a ser chamada de renegociação insegura e deve ser desativada.

⚠️ O TLS 1.3 eliminou completamente o mecanismo de renegociação, substituindo-o por um processo mais seguro chamado Key Update, que não apresenta essa vulnerabilidade.

Como é verificado?

O teste verifica se o servidor web:

- Suporta a renegociação segura, com o vínculo criptográfico definido pela RFC 5746
- Rejeita tentativas de renegociação insegura, sem o vínculo criptográfico

Riscos do não atendimento

Risco	Consequência
SSL Renegotiation Attack	O invasor intercepta a conexão TLS e injeta tráfego malicioso antes de repassá-la ao servidor legítimo, que interpreta o tráfego injetado como parte da sessão autenticada do cliente
Session Splicing	O invasor divide a sessão em partes, inserindo comandos maliciosos entre segmentos legítimos da comunicação
Session Interception	O invasor assume o controle de uma sessão já estabelecida, aproveitando a ausência de vínculo entre os handshakes

Como resolver?

- Ativar a extensão de renegociação segura (RFC 5746) no servidor web;
- Desativar explicitamente a renegociação insegura, rejeitando clientes que não suportem a extensão segura;
- Servidores que utilizam TLS 1.3 exclusivamente já estão protegidos por padrão, pois o protocolo não suporta renegociação no modelo antigo.

Comparativo: renegociação no TLS 1.2 × TLS 1.3

Característica	TLS 1.2	TLS 1.3
Mecanismo de atualização	Renegociação	Key Update
Vínculo entre handshakes	Opcional (RFC 5746)	Não se aplica
Vulnerabilidade CVE-2009-3555	Presente sem RFC 5746	Eliminada por design
Configuração necessária	Habilitar a extensão da RFC 5746 e rejeitar a renegociação insegura, por meio da diretiva própria do servidor web em uso	Nenhuma
Risco de DoS por renegociação	Presente	Eliminado por design

> 💡 **Dica:** A melhor mitigação para vulnerabilidades de renegociação é a adoção do TLS 1.3 exclusivo (seção 3.1 deste manual), que elimina o mecanismo de renegociação por design. As configurações acima são necessárias principalmente para servidores que mantêm suporte ao TLS 1.2 por compatibilidade com clientes mais antigos.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que habilitam a renegociação segura e rejeitam a renegociação insegura para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre renegociação TLS, com explicação da vulnerabilidade e recomendações de configuração	wiki.mozilla.org/Security/Server_Side_TLS
RFC 5746	Especificação da extensão de renegociação segura (Renegotiation Indication Extension)	datatracker.ietf.org/doc/html/rfc5746
RFC 8446	Especificação do TLS 1.3, que eliminou o mecanismo de renegociação por design, substituindo-o pelo Key Update	datatracker.ietf.org/doc/html/rfc8446
IETF RFC 7457	Documento que sumariza os ataques conhecidos contra TLS, incluindo ataques de renegociação insegura	datatracker.ietf.org/doc/html/rfc7457
SSL Labs — SSL Test	Verifica se o servidor suporta renegociação segura e rejeita renegociação insegura, com itens específicos no relatório	ssllabs.com/ssltest

💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) verifica explicitamente os itens "Secure Renegotiation" — que deve indicar "Supported" — e "Insecure Renegotiation" — que deve indicar "No" — no relatório de análise do servidor.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A configuração da renegociação segura é feita nos arquivos de configuração do servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo

- A maioria dos servidores web e bibliotecas TLS modernas já implementa a renegociação segura por padrão, exigindo apenas a verificação e, eventualmente, a desativação explícita da versão insegura.
- Servidores que já utilizam TLS 1.3 exclusivamente não precisam de nenhuma ação adicional.
- O principal ponto de atenção é garantir que clientes legítimos não dependam da renegociação insegura, situação cada vez mais rara em ambientes atualizados.

🛡️ Benefício estimado: Médio

- A exploração dessa vulnerabilidade exige que o invasor esteja posicionado entre o cliente e o servidor, o que não é trivial em redes bem segmentadas.
- No entanto, a vulnerabilidade é conhecida desde 2009 (CVE-2009-3555) e possui exploits documentados, tornando-a acessível a invasores com conhecimento intermediário.
- A adoção crescente do TLS 1.3 reduz progressivamente o universo de sistemas vulneráveis, mas servidores que ainda suportam TLS 1.2 permanecem expostos enquanto a renegociação insegura não for desativada.

3.8 Renegociação Iniciada pelo Cliente

O que é?

Como visto no controle 3.7 (Renegociação Segura), o TLS permite a renegociação de parâmetros criptográficos durante uma sessão já estabelecida. Essa renegociação pode ser iniciada tanto pelo servidor quanto pelo cliente.

Permitir que o cliente inicie a renegociação é, na maioria dos casos, desnecessário e abre uma superfície de ataque específica: como o processo de renegociação é computacionalmente custoso para o servidor, um invasor pode explorar esse recurso para sobrecarregá-lo deliberadamente.

> 💡 **Analogia:** imagine que cada vez que um cliente pede para "reiniciar" a conversa segura, o servidor precisa resolver um problema matemático complexo. Um invasor pode ficar pedindo reinicializações repetidamente, esgotando os recursos do servidor sem precisar de muitas conexões simultâneas.

> ⚠️ O TLS 1.3 eliminou completamente o mecanismo de renegociação, tornando este controle relevante apenas para servidores que ainda suportam TLS 1.2 ou versões anteriores.

Como é verificado?

O teste verifica se o servidor web aceita solicitações de renegociação iniciadas pelo cliente.

Riscos do não atendimento

Risco	Consequência
DoS por renegociação	O atacante estabelece uma conexão TLS legítima e emite requisições de renegociação de forma contínua, exaurindo os recursos do servidor desproporcionalmente. Nesse cenário, uma única conexão maliciosa demanda processamento equivalente a centenas de conexões padrão

Risco	Consequência
Man-in-the-Middle (MitM)	A renegociação iniciada pelo cliente pode ser explorada em conjunto com a renegociação insegura — ver controle 3.7 (Renegociação Segura) — para injetar tráfego malicioso na sessão

> 💡 **Comparação com outros ataques DoS:** ataques DoS convencionais, que abrem muitas conexões simultâneas, são mais fáceis de detectar e bloquear por ferramentas de mitigação padrão. A renegociação iniciada pelo cliente é mais insidiosa porque o ataque ocorre dentro de uma única conexão legítima e já estabelecida, dificultando sua detecção.

Como resolver?

- Desabilitar a renegociação iniciada pelo cliente no servidor web;
- Servidores que utilizam TLS 1.3 exclusivamente já estão protegidos por padrão, pois o protocolo não suporta renegociação.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que desabilitam a renegociação iniciada pelo cliente para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre renegociação iniciada pelo cliente, com explicação dos riscos e recomendações de configuração	wiki.mozilla.org/Security/Server_Side_TLS
RFC 5746	Especificação da extensão de renegociação segura, base para entendimento da diferença entre renegociação segura e insegura	datatracker.ietf.org/doc/html/rfc5746
RFC 8446	Especificação do TLS 1.3, que eliminou o mecanismo de renegociação por design	datatracker.ietf.org/doc/html/rfc8446

Referência	Descrição	URL
OWASP — TLS Cheat Sheet	Guia prático da OWASP sobre configuração segura de TLS, incluindo recomendações sobre renegociação iniciada pelo cliente	cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html
SSL Labs — SSL Test	Verifica se o servidor aceita renegociação iniciada pelo cliente, com item específico "Client-initiated Renegotiation" no relatório — deve indicar "No"	ssllabs.com/ssltest

> 💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) verifica explicitamente o item "Client-initiated Renegotiation" no relatório de análise do servidor, que deve indicar "No". A adoção do TLS 1.3 exclusivo (seção 3.1 deste manual) elimina a necessidade desta configuração por completo, sendo a solução mais robusta e de longo prazo para ambos os controles 3.7 (Renegociação Segura) e 3.8 (Renegociação Iniciada pelo Cliente).

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A desativação da renegociação iniciada pelo cliente é feita por configuração no servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à desativação de um parâmetro no servidor web, bem documentada para os principais servidores (Apache, Nginx, IIS).
- Na prática, poucos cenários legítimos dependem da renegociação iniciada pelo cliente, minimizando o risco de impacto em funcionalidades existentes.
- Servidores que já utilizam TLS 1.3 exclusivamente não precisam de nenhuma ação adicional.

🛡️ Benefício estimado: Médio

- O ataque de DoS por renegociação exige que o invasor já tenha estabelecido uma conexão TLS legítima com o servidor, o que representa uma barreira inicial.
- No entanto, uma vez estabelecida a conexão, o ataque é de baixo custo para o invasor e de alto custo para o servidor, tornando-o assimétrico e potencialmente eficaz mesmo com poucos recursos.
- A combinação de custo nulo e proteção contra um vetor de ataque específico e pouco mitigável por outros meios torna o controle claramente vantajoso.

3.9 0-RTT (Zero Round Trip Time)

O que é?

O 0-RTT é um recurso introduzido no TLS 1.3 que permite ao cliente enviar dados para o servidor antes mesmo de o handshake ser concluído, aproveitando uma chave pré-compartilhada (PSK) de uma sessão anterior. O objetivo é reduzir a latência em reconexões, melhorando o desempenho.

> 💡 **Analogia:** é como um cliente assíduo de um banco que, por ser reconhecido pelo segurança, entra e já faz sua solicitação antes de apresentar os documentos formalmente. Essa agilidade é conveniente, mas cria uma vulnerabilidade: um fraudador poderia imitar esse cliente e repetir a mesma solicitação sem ser percebido.

Em uma conexão TLS convencional, são necessárias pelo menos duas trocas de mensagens (round trips) antes que os dados possam ser enviados. O 0-RTT elimina essa espera, mas ao custo de uma propriedade de segurança importante: a proteção contra repetição de requisições.

Modo	Round Trips antes de enviar dados
TLS 1.2 convencional	2

Modo	Round Trips antes de enviar dados
TLS 1.3 convencional	1
TLS 1.3 com 0-RTT	0

⚠️ O NIST SP 800-52 Rev. 2 recomenda que, em geral, servidores não devem suportar 0-RTT. Caso seja indispensável por razões de desempenho, deve-se adotar o mecanismo de Single-Use Tickets, definido na RFC 8446, que invalida cada ticket após o primeiro uso.

Como é verificado?

O teste verifica se o servidor web oferece suporte ao modo 0-RTT. O resultado esperado é que esse suporte não esteja disponível.

Riscos da presença do 0-RTT

Risco	Consequência
Replay Attack	Um invasor captura os dados enviados pelo cliente no modo 0-RTT e os retransmite ao servidor posteriormente. Como esses dados chegam antes da autenticação completa, o servidor pode processá-los novamente como se fossem uma nova requisição legítima, gerando ações duplicadas não autorizadas

💡 **Exemplo prático de Replay Attack:** imagine que um usuário envia uma transferência bancária no modo 0-RTT. Um invasor que capturou esse pacote pode retransmiti-lo múltiplas vezes, fazendo o servidor processar várias transferências do mesmo valor, sem que o usuário tenha solicitado.

⚠️ O Replay Attack no 0-RTT é especialmente perigoso em operações não idempotentes, ou seja, operações cujo efeito muda a cada execução, como transações financeiras, envio de formulários e alterações de dados.

Como resolver?

- Desativar o suporte a 0-RTT no servidor web, que é a recomendação do NIST para a maioria dos casos;
- Caso o 0-RTT seja indispensável por razões de desempenho, implementar o mecanismo de Single-Use Tickets (RFC 8446, seção 8.1), que invalida cada ticket após o primeiro uso, mitigando a retransmissão.

Comparativo: desempenho × segurança

Modo	Round trips	Risco de Replay Attack	Recomendação
TLS 1.2 convencional	2	Nenhum	⚠️ Manter apenas por compatibilidade
TLS 1.3 convencional	1	Nenhum	✅ Recomendado
TLS 1.3 com 0-RTT desabilitado	1	Nenhum	✅ Recomendado
TLS 1.3 com 0-RTT habilitado	0	Alto em operações não idempotentes	❌ Evitar

Fluxo de decisão para uso do 0-RTT

```
O serviço processa operações não idempotentes?
(formulários, transações, alterações de dados)
│
├── SIM → Desabilitar 0-RTT ✅
│
└── NÃO → O ganho de desempenho justifica o risco?
          │
          ├── NÃO → Desabilitar 0-RTT ✅
          │
```

```
└── SIM → Habilitar 0-RTT com Single-Use Tickets
                e tratamento explícito na aplicação ⚠️
```

⚠️ **Atenção:** mesmo com 0-RTT habilitado e Single-Use Tickets configurados, a proteção contra Replay Attack depende da aplicação tratar corretamente as requisições de dados antecipados. A configuração no servidor é necessária, mas não suficiente. A equipe de desenvolvimento da aplicação deve estar ciente dessa responsabilidade.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que desabilitam o 0-RTT para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre 0-RTT, com explicação dos riscos e recomendações de configuração	wiki.mozilla.org/Security/Server_Side_TLS
RFC 8446 — Seção 8.1	Especificação do mecanismo de Single-Use Tickets para mitigação de Replay Attack no 0-RTT	datatracker.ietf.org/doc/html/rfc8446#section-8.1
NIST SP 800-52 Rev. 2	Diretrizes do NIST sobre 0-RTT, com recomendação explícita de desabilitação na maioria dos casos	csrc.nist.gov/pubs/sp/800/52/r2/final
OWASP — TLS Cheat Sheet	Guia prático da OWASP sobre configuração segura de TLS, incluindo recomendações sobre 0-RTT e proteção contra Replay Attack	cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html
SSL Labs — SSL Test	Verifica se o servidor oferece suporte a 0-RTT, com item específico no relatório — deve indicar "No"	ssllabs.com/ssltest

💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) verifica explicitamente o item "0-RTT" no relatório de análise do servidor, que deve indicar "No". Para serviços públicos ao cidadão, a desabilitação do 0-RTT é a escolha correta na grande maioria dos casos, dado que praticamente todos os serviços governamentais processam operações não idempotentes — formulários, autenticações, consultas com registro de acesso e transações de dados do cidadão.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A desativação do 0-RTT é feita por configuração no servidor web já existente, sem necessidade de aquisição de produtos, licenças ou serviços adicionais.

🔧 Custo operacional estimado: Baixo a Médio

- A desativação em si é simples, resumindo-se à alteração de um parâmetro de configuração.
- No entanto, caso o 0-RTT esteja sendo utilizado para ganho de desempenho em aplicações críticas, sua desativação pode exigir uma avaliação de impacto e eventual redesenho de componentes que dependiam dessa otimização.
- Caso a opção seja manter o 0-RTT com Single-Use Tickets, a implementação e validação desse mecanismo exige maior esforço técnico e testes mais cuidadosos.

🛡️ Benefício estimado: Médio a Alto

- O Replay Attack via 0-RTT exige que o invasor consiga capturar o pacote inicial do cliente, o que demanda posicionamento privilegiado na rede.
- No entanto, o impacto de um ataque bem-sucedido pode ser severo em aplicações que processam operações não idempotentes, como transações financeiras ou alterações de dados críticos.
- A desativação do 0-RTT é simples e sem custo financeiro, tornando a relação custo-benefício favorável à sua desativação na maioria dos cenários.

3.10 OCSP Stapling

O que é?

Quando um usuário acessa um site via HTTPS, o navegador precisa verificar se o certificado digital do servidor ainda é válido, ou seja, se não foi revogado pela autoridade certificadora (AC). Existem dois mecanismos tradicionais para isso:

Mecanismo	Como funciona	Problema
CRL (Lista de Revogação)	O navegador baixa uma lista completa de certificados revogados	Lista grande, lenta e pesada
OCSP (Protocolo de Status Online)	O navegador consulta diretamente a AC para verificar o certificado específico	Expõe o histórico de navegação do usuário à AC e adiciona latência ao handshake

O OCSP Stapling resolve esses problemas: em vez de o navegador consultar a AC, o próprio servidor busca periodicamente a resposta OCSP junto à AC e a "grampeia" (staple) no handshake TLS, entregando-a diretamente ao cliente.

> 💡 **Analogia:** em vez de o cliente ligar para a prefeitura para verificar se o alvará de funcionamento de um estabelecimento é válido, o próprio estabelecimento exibe o alvará atualizado na entrada, assinado pela prefeitura e com data de validade recente. O cliente não precisa consultar ninguém.

Como é verificado?

O teste verifica se o servidor web inclui a resposta OCSP no handshake TLS, ou seja, se o OCSP Stapling está habilitado.

Riscos da ausência do OCSP Stapling

Risco	Consequência
Exposição de privacidade	Sem OCSP Stapling, o navegador consulta diretamente a AC a cada nova conexão, revelando quais sites o usuário está acessando e quando
Latência adicional	A consulta à AC adiciona um passo extra ao handshake TLS, aumentando o tempo de carregamento do site, especialmente quando o servidor da AC está lento ou indisponível
Falha silenciosa	Em alguns navegadores, se a consulta OCSP falhar, a conexão é estabelecida mesmo assim (soft-fail), tornando a verificação de revogação ineficaz

Como resolver?

Habilitar o OCSP Stapling no servidor web. O servidor passará a buscar e armazenar em cache a resposta OCSP periodicamente, entregando-a diretamente ao cliente durante o handshake, sem necessidade de consulta externa pelo navegador.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações com OCSP Stapling habilitado para os principais servidores web	ssl-config.mozilla.org
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre OCSP Stapling, com explicação do mecanismo e recomendações de configuração	wiki.mozilla.org/Security/Server_Side_TLS
RFC 6960	Especificação técnica do protocolo OCSP (Online Certificate Status Protocol)	datatracker.ietf.org/doc/html/rfc6960
RFC 6066	Especificação das extensões TLS, incluindo a extensão Certificate Status Request, que define o mecanismo de OCSP Stapling	datatracker.ietf.org/doc/html/rfc6066#section-8

Referência	Descrição	URL
Let's Encrypt — Ending OCSP Support	Anúncio oficial da descontinuação do serviço OCSP pela Let's Encrypt, com o cronograma de desativação e a migração para listas de revogação (CRL)	letsencrypt.org/2024/12/05/ending-ocsp
SSL Labs — SSL Test	Verifica se o servidor oferece OCSP Stapling, com item específico "OCSP stapling" no relatório — deve indicar "Yes"	ssllabs.com/ssltest

💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) verifica explicitamente o item "OCSP stapling" no relatório de análise do servidor, que deve indicar "Yes". Após habilitar o OCSP Stapling, pode ser necessário aguardar alguns minutos para que o servidor busque e armazene em cache a primeira resposta OCSP junto à AC antes que o SSL Labs consiga verificá-lo corretamente.

⚠️ **Atenção:** o OCSP Stapling depende de o servidor conseguir acessar o endpoint OCSP da AC. Em ambientes com restrições de saída de rede (firewalls, proxies), é necessário garantir que o servidor tenha acesso ao endereço OCSP indicado no certificado, geralmente listado na extensão "Authority Information Access" do certificado digital.

⚠️ **Atenção — o OCSP está sendo descontinuado por parte das ACs:** pelos mesmos motivos de privacidade descritos acima, algumas autoridades certificadoras deixaram de operar servidores OCSP e passaram a publicar apenas listas de revogação (CRL). A Let's Encrypt, por exemplo, removeu o endereço OCSP dos certificados emitidos a partir de maio de 2025 e desligou seus servidores OCSP em agosto de 2025.

Na prática, isso significa que:

• Certificados sem endereço OCSP **não permitem** habilitar o OCSP Stapling — não se trata de erro de configuração, mas de ausência do serviço na AC emissora. Nesses

casos, a verificação de revogação passa a ser feita pelos mecanismos próprios dos navegadores, alimentados por CRLs.

• O controle permanece plenamente aplicável aos certificados de ACs que ainda mantêm o serviço OCSP, situação comum em certificados OV e EV comerciais e no âmbito da ICP-Brasil.

• Antes de investir esforço na configuração, verifique se o certificado em uso possui endereço OCSP na extensão "Authority Information Access".

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- O OCSP Stapling é um recurso nativo dos principais servidores web (Apache, Nginx, IIS) e não requer aquisição de produtos, licenças ou serviços adicionais. As respostas OCSP são fornecidas sem custo pelas autoridades certificadoras que mantêm esse serviço.

🔧 Custo operacional estimado: Baixo

- A habilitação resume-se à adição de alguns parâmetros nos arquivos de configuração do servidor web, com documentação amplamente disponível.
- O servidor gerencia automaticamente a renovação periódica das respostas OCSP em cache, sem intervenção manual contínua.
- O principal ponto de atenção é garantir que o servidor tenha acesso à URL do servidor OCSP da AC, o que pode exigir ajustes em regras de firewall em ambientes mais restritivos.

🛡️ Benefício estimado: Médio a Alto

- O OCSP Stapling não protege contra ataques diretos ao servidor ou ao usuário, mas melhora significativamente a privacidade e o desempenho da conexão.
- A proteção de privacidade é relevante especialmente em sites governamentais e de saúde, onde o simples fato de o usuário acessar o site pode ser uma informação sensível.
- A melhoria de desempenho é um benefício adicional concreto: eliminar a consulta externa à AC reduz a latência do handshake TLS, melhorando a experiência do usuário.

- A facilidade de implementação e o custo nulo tornam a relação custo-benefício claramente favorável.

3.11 Cadeia de Confiança do Certificado

O que é?

Quando um navegador acessa um site via HTTPS, ele precisa verificar se o certificado digital apresentado pelo servidor é legítimo. Essa verificação é feita por meio da cadeia de confiança, uma hierarquia de certificados que liga o certificado do site a uma Autoridade Certificadora (AC) raiz, reconhecida e confiável pelos navegadores.

💡 **Analogia:** é como verificar a autenticidade de um documento. O documento é assinado por um cartório, que por sua vez é credenciado por um órgão estadual, que é reconhecido pelo governo federal. Se qualquer elo dessa cadeia for inválido ou estiver faltando, a autenticidade não pode ser confirmada.

⚠️ Certificados autoassinados, nos quais o próprio servidor assina seu certificado sem uma AC reconhecida, não constroem uma cadeia de confiança válida e são bloqueados pelos navegadores com alertas de segurança, além de serem vulneráveis a ataques MitM.

Como é verificado?

O teste verifica se é possível construir uma cadeia de confiança válida e completa para o certificado do site, partindo do certificado do servidor até uma AC raiz reconhecida.

Riscos da ausência de cadeia de confiança válida

Risco	Consequência
Ataque Man-in-the-Middle (MitM)	Sem uma cadeia de confiança válida, o navegador não pode confirmar a identidade do servidor. Um invasor pode apresentar um certificado falso ou autoassinado e o usuário não terá como distingui-lo de um legítimo, especialmente se ignorar os alertas do navegador
Bloqueio pelo navegador	Navegadores modernos bloqueiam ou exibem alertas críticos para sites com cadeia de confiança inválida, comprometendo o acesso legítimo dos usuários
Perda de confiança	Usuários que se deparam com alertas de certificado tendem a abandonar o site ou, pior, a desenvolver o hábito de ignorar alertas de segurança

Como resolver?

- Utilizar um certificado emitido por uma AC reconhecida pelos principais navegadores
- Configurar o servidor web para fornecer não apenas o certificado do site, mas também todos os certificados intermediários da cadeia, garantindo que o navegador consiga validar o caminho completo até a AC raiz
- Nunca utilizar certificados autoassinados em ambientes de produção acessíveis ao público

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla SSL Configuration Generator	Gera automaticamente configurações que incluem a cadeia de certificados completa para os principais servidores web	ssl-config.mozilla.org
Mozilla — Chain of Trust	Documentação da Mozilla sobre cadeia de confiança de certificados, com explicação do mecanismo de validação pelos navegadores	developer.mozilla.org/en-US/docs/Web/Security/Certificate_Transparency

Referência	Descrição	URL
ITI — ICP-Brasil	Infraestrutura de Chaves Públicas Brasileira. AC raiz oficial para certificados digitais no Brasil, incluindo certificados para sites governamentais	iti.gov.br
Let's Encrypt — Chain of Trust	Documentação da Let's Encrypt sobre sua cadeia de confiança, com informações sobre certificados intermediários e raízes reconhecidas	letsencrypt.org/certificates
SSL Labs — SSL Test	Verifica a cadeia de confiança do certificado do servidor, identificando certificados intermediários ausentes e problemas de validação	ssllabs.com/ssltest
RFC 5280	Especificação técnica do formato de certificados X.509 e do mecanismo de validação de cadeia de confiança	datatracker.ietf.org/doc/html/rfc5280

💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) verifica a cadeia de confiança completa do servidor na seção "Certification Paths" do relatório, identificando certificados intermediários ausentes e eventuais problemas de validação. Um erro comum é configurar o servidor apenas com o certificado do domínio, esquecendo de incluir os certificados intermediários da AC — o que resulta em falha de validação em alguns clientes, mesmo que o certificado em si seja válido.

⚠️ **Atenção para órgãos públicos:** certificados emitidos pela ICP-Brasil são reconhecidos como padrão para documentos digitais no âmbito da administração pública federal brasileira. Para sites institucionais, verificar junto à área jurídica e de TI se há exigência normativa de uso de certificados ICP-Brasil, especialmente em sistemas que envolvam assinatura digital, autenticação de usuários ou transações com valor legal.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo

- Certificados emitidos por ACs reconhecidas podem ser gratuitos (ex.: Let's Encrypt) ou pagos, dependendo do nível de validação necessário.
- Certificados DV (Domain Validation) estão amplamente disponíveis de forma gratuita e são tecnicamente suficientes para construir uma cadeia de confiança válida. Para sites institucionais de órgãos públicos, contudo, aplica-se a ressalva do controle 2.1 (HTTPS Disponível): por não atestarem a identidade da organização, os certificados DV devem ser tratados como solução temporária.
- Certificados OV (Organization Validation) ou EV (Extended Validation), que oferecem maior nível de confiança e são mais indicados para sites institucionais, têm custo anual que varia de algumas centenas a alguns milhares de reais, dependendo da AC e do número de domínios cobertos.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à instalação correta do certificado e dos certificados intermediários no servidor web, processo bem documentado para os principais servidores.
- Ferramentas como o SSL Labs permitem verificar rapidamente se a cadeia está completa e corretamente configurada.
- O principal ponto de atenção é o monitoramento da validade do certificado e sua renovação antes do vencimento, processo que pode ser automatizado com ferramentas como o Certbot (para certificados Let's Encrypt).

🛡️ Benefício estimado: Alto

- A cadeia de confiança é o fundamento da identidade digital do servidor. Sem ela, todos os demais controles de segurança TLS perdem eficácia, pois o usuário não tem como confirmar que está se comunicando com o servidor legítimo.
- Um certificado inválido ou ausente compromete todos os usuários do site, não apenas casos isolados.

- O impacto de um ataque MitM bem-sucedido por ausência de cadeia de confiança pode incluir roubo de credenciais, dados pessoais e informações sensíveis em larga escala.

3.12 Chave Pública do Certificado

O que é?

O certificado digital de um site contém uma chave pública utilizada para verificar a identidade do servidor e estabelecer a comunicação criptografada. A segurança dessa chave depende de dois fatores:

- O algoritmo utilizado para gerar a assinatura digital
- O tamanho ou os parâmetros da chave, que determinam a resistência a ataques

Os principais algoritmos utilizados em certificados digitais são:

Algoritmo	Tipo	Tamanho mínimo recomendado
RSA	Chave assimétrica baseada em fatoração	2048 bits
ECDSA	Chave baseada em curvas elípticas	224 bits (curvas NIST)
EdDSA	Assinatura digital baseada em curvas de Edwards	Curvas Ed25519 ou Ed448

💡 **Analogia:** é como o tamanho da combinação de um cofre. Uma combinação de 3 dígitos pode ser descoberta por força bruta em segundos; uma de 20 dígitos exigiria tempo astronômico. O algoritmo e o tamanho da chave pública determinam o esforço computacional necessário para que um invasor descubra a chave privada correspondente. Quanto maiores e mais robustos os parâmetros, mais segura é a identidade do servidor.

⚠️ Chaves de tamanho insuficiente ou baseadas em parâmetros inadequados podem ser revertidas com poder computacional crescente, permitindo que um invasor obtenha a chave privada do servidor e falsifique sua identidade. O algoritmo de hash utilizado para assinar o certificado é um aspecto distinto, tratado no item 3.13 deste manual.

Como é verificado?

O teste verifica se a chave pública do certificado, independentemente de ser RSA, ECDSA ou EdDSA, utiliza algoritmo e parâmetros de tamanho considerados seguros pelas diretrizes do NIST e da OWASP.

Riscos do não atendimento

Risco	Consequência
Falsificação de identidade	Chaves curtas ou algoritmos obsoletos podem ser quebrados, permitindo que um invasor forje o certificado do servidor e se passe por ele perante os usuários
Ataques side-channel	Curvas elípticas inadequadas expõem a chave privada a ataques que exploram informações indiretas do processamento criptográfico, como tempo de execução ou padrões de acesso à memória
Comprometimento retroativo	Uma chave fraca que foi quebrada pode comprometer sessões passadas que foram gravadas pelo invasor, especialmente em conexões sem sigilo futuro (Forward Secrecy)

Como resolver?

- **Para RSA**: gerar o certificado com chave de no mínimo **2048 bits** (seguro até 2030 segundo o NIST); para certificados com validade superior a dois anos, preferir **3072 bits** — RSA 4096 bits adiciona overhead computacional significativo (~6× em relação ao RSA 2048) com benefício prático limitado dado o ciclo de renovação habitual de 1–2 anos, sendo preferível migrar para ECDSA

- **Para ECDSA**: utilizar curvas recomendadas pelo NIST SP 800-186 (ex.: **P-256**, **P-384**), que oferecem segurança equivalente ao RSA com chaves significativamente menores
- **Para EdDSA**: utilizar as curvas **Ed25519** ou **Ed448**, conforme a RFC 8032, que oferecem resistência nativa a ataques *side-channel* por serem implementadas com operações de tempo constante
- **Renovação de certificados**: renovar certificados com parâmetros obsoletos mesmo antes do vencimento, caso não atendam aos critérios mínimos — ao renovar, sempre **gerar um novo par de chaves** em vez de reutilizar o anterior

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — Server Side TLS	Documentação técnica da Mozilla sobre algoritmos e tamanhos de chave recomendados para certificados digitais	wiki.mozilla.org/Security/Server_Side_TLS
NIST SP 800-186	Recomendações do NIST para uso de curvas elípticas em criptografia, incluindo lista de curvas aprovadas para uso em certificados	csrc.nist.gov/publications/detail/sp/800-186/final
NIST SP 800-57 Part 1	Diretrizes do NIST para gestão de chaves criptográficas, incluindo recomendações de tamanho mínimo de chave por algoritmo e horizonte temporal	csrc.nist.gov/pubs/sp/800/57/pt1/r5/final
NIST SP 800-131A Rev. 2	Diretrizes do NIST para transição de algoritmos criptográficos, incluindo descontinuação de RSA com chaves curtas e SHA-1	csrc.nist.gov/publications/detail/sp/800-131a/rev-2/final
RFC 8032	Especificação do EdDSA e das curvas de assinatura Ed25519 e Ed448, recomendadas por sua resistência a ataques side-channel	datatracker.ietf.org/doc/html/rfc8032

Referência	Descrição	URL
OWASP — TLS Cheat Sheet	Guia prático da OWASP sobre algoritmos e tamanhos de chave recomendados para certificados digitais	cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html
SSL Labs — SSL Test	Verifica o algoritmo e o tamanho da chave do certificado do servidor, com avaliação de conformidade com as boas práticas	ssllabs.com/ssltest
ITI — ICP-Brasil	Requisitos técnicos para emissão de certificados digitais no âmbito da ICP-Brasil, incluindo algoritmos e tamanhos de chave aceitos	iti.gov.br

💡 **Dica:** o SSL Labs (ssllabs.com/ssltest) exibe na seção "Certificate" do relatório os campos "Key" e "Public Key", que informam o algoritmo e o tamanho da chave pública do certificado. Um certificado RSA de 2048 bits atende ao mínimo recomendado; certificados ECDSA com P-256 oferecem segurança equivalente com chaves significativamente menores, resultando em handshakes mais rápidos — vantagem relevante em serviços públicos com alto volume de acessos. Para verificar o algoritmo de hash utilizado na assinatura do certificado, consulte o controle 3.13 deste manual.

⚠️ **Atenção para órgãos públicos:** a ICP-Brasil define os algoritmos e tamanhos de chave aceitos para certificados digitais com validade legal no Brasil. Ao solicitar ou renovar certificados para sistemas que exijam conformidade com a ICP-Brasil, verificar os requisitos técnicos vigentes junto ao ITI (iti.gov.br), pois podem diferir parcialmente das recomendações internacionais do NIST e da Mozilla.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- A escolha dos parâmetros da chave é feita no momento da **geração do certificado**, sem custo adicional em relação à emissão de um certificado com parâmetros fracos.

- Caso o certificado atual utilize parâmetros inadequados, será necessário **revogá-lo e emitir um novo**, o que pode envolver custo de reemissão dependendo da autoridade certificadora. Contudo, muitas ACs oferecem **reemissão gratuita**, e certificados **Let's Encrypt** não têm custo algum.
- O uso de chaves maiores, como **RSA 4096 bits**, pode impactar levemente o desempenho do servidor, mas não gera custo financeiro direto.

🔧 Custo operacional estimado: Baixo

- A geração de um certificado com **parâmetros seguros** segue o mesmo processo de um certificado convencional, com a simples especificação dos parâmetros corretos no momento da **criação do par de chaves**.
- Ferramentas gratuitas como **SSL Labs** permitem verificar rapidamente se o certificado atual atende aos requisitos mínimos.
- O principal esforço ocorre **apenas uma vez**: na substituição de um certificado com parâmetros inadequados, após o qual a manutenção segue o ciclo normal de renovação.

🛡️ Benefício estimado: Médio ou Alto

- A quebra de chaves **RSA de 2048 bits** ou de **curvas elípticas** recomendadas ainda exige poder computacional muito elevado, tornando ataques práticos pouco prováveis no curto prazo.
- No entanto, o avanço da **computação quântica** representa uma ameaça crescente a algoritmos baseados em fatoração e curvas elípticas, tornando a escolha de **parâmetros robustos** desde já uma medida de proteção com horizonte de longo prazo.
- A combinação de custo baixo com proteção contra **falsificação de identidade**, um dos ataques de maior impacto para os usuários, torna o controle claramente vantajoso.

3.13 Assinatura do Certificado

O que é?

Todo certificado digital é assinado digitalmente pela Autoridade Certificadora (AC) que o emitiu, garantindo sua autenticidade e integridade. Essa assinatura é gerada com uma função

hash, que produz uma "impressão digital" do certificado.

Se a função hash utilizada for fraca ou obsoleta, essa impressão digital pode ser falsificada, permitindo que um invasor forge um certificado aparentemente legítimo.

ℹ️ **Relação com o controle 3.5 (Função Hash para Troca de Chaves):** enquanto o controle 3.5 (Função Hash para Troca de Chaves) trata das funções hash usadas durante o handshake TLS para autenticar mensagens em tempo real, este controle trata da função hash usada pela AC para assinar o próprio certificado, que é uma operação prévia e permanente, válida por toda a vida útil do certificado.

O status atual dos principais algoritmos de hash para assinatura de certificados é:

Algoritmo	Status	Recomendação
MD5	Quebrado	❌ **Proibido**
SHA-1	Vulnerável	❌ **Descontinuado, apenas para sistemas legados**
SHA-256	Seguro	✅ **Mínimo recomendado**
SHA-384 / SHA-512	Seguro	✅ **Recomendado para maior longevidade**

💡 **Analogia:** é como o lacre de autenticidade em um medicamento. O lacre garante que o produto é genuíno e não foi adulterado — mas apenas se for difícil de imitar. Se o lacre puder ser copiado facilmente, qualquer pessoa pode embalar um produto falsificado como se fosse original. A função hash na assinatura do certificado cumpre esse papel: permite que qualquer cliente verifique que o certificado foi emitido por uma AC legítima — mas somente se o algoritmo utilizado for resistente a colisões.

Como é verificado?

O teste verifica se o certificado do site foi assinado pela AC utilizando um algoritmo de hash seguro, sendo SHA-256 o mínimo aceitável.

Riscos do não atendimento

O uso de algoritmos obsoletos na assinatura do certificado expõe o servidor aos mesmos ataques descritos no controle 3.5 (Função Hash para Troca de Chaves), com um agravante: como a assinatura é parte permanente do certificado, o comprometimento é estrutural e persistente até a substituição do certificado.

Risco	Consequência
Falsificação de certificado	Um invasor explora colisões no MD5 ou SHA-1 para forjar um certificado com a mesma assinatura de um legítimo, personificando o servidor perante os usuários
Ataque MitM	Com um certificado forjado em mãos, o invasor pode interceptar e adulterar toda a comunicação entre o usuário e o servidor sem que o navegador emita alertas
Bloqueio pelos navegadores	Certificados assinados com MD5 ou SHA-1 já são rejeitados pelos principais navegadores modernos, tornando o site inacessível para a maioria dos usuários

Como resolver?

- **Verificar o algoritmo de assinatura** do certificado atual, disponível nos detalhes do certificado em qualquer navegador ou por ferramentas como o **SSL Labs**
- Caso o certificado utilize **MD5 ou SHA-1**, **revogá-lo e emitir um novo** assinado com **SHA-256 ou superior**
- Ao renovar ou emitir novos certificados, especificar explicitamente o uso de **SHA-256**, **SHA-384** ou **SHA-512**

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — Forbidden and Deprecated Practices	Documentação da Mozilla sobre algoritmos proibidos e descontinuados para assinatura de certificados, incluindo MD5 e SHA-1	wiki.mozilla.org/Security/Server_Side_TLS
CA/Browser Forum — Baseline Requirements	Requisitos técnicos mínimos que todas as ACs públicas devem seguir para emissão de certificados, incluindo algoritmos de assinatura aceitos	cabforum.org/working-groups/server/baseline-requirements
NIST SP 800-131A Rev. 2	Diretrizes do NIST para transição de algoritmos criptográficos, incluindo a descontinuação formal do SHA-1 e MD5 para assinatura de certificados	csrc.nist.gov/publications/detail/sp/800-131a/rev-2/final
NIST FIPS 180-4	Especificação oficial das funções hash SHA-2 aprovadas pelo NIST para uso em assinaturas digitais	csrc.nist.gov/publications/detail/fips/180/4/final
SHAttered — Ataque ao SHA-1	Documentação técnica do ataque SHAttered, que demonstrou a viabilidade prática de colisões no SHA-1 em certificados digitais	marc-stevens.nl/research/shattered.io/
SSL Labs — SSL Test	Verifica o algoritmo de assinatura do certificado do servidor na seção "Certificate" do relatório — o campo "Signature algorithm" não deve conter MD5 ou SHA1	ssllabs.com/ssltest
ITI — ICP-Brasil	Requisitos técnicos para assinatura de certificados digitais no âmbito da ICP-Brasil, incluindo algoritmos de hash aceitos	iti.gov.br

> 💡 **Dica:** o algoritmo de assinatura do certificado pode ser verificado diretamente no navegador, clicando no cadeado na barra de endereços e acessando os detalhes

do certificado. O campo "Algoritmo de Assinatura" deve indicar SHA-256, SHA-384 ou SHA-512. Qualquer referência a MD5 ou SHA-1 exige substituição imediata do certificado.

⚠️ **Atenção:** diferentemente de outras configurações deste manual, que podem ser ajustadas no servidor web, a função hash de assinatura do certificado é definida no momento da emissão pela AC e não pode ser alterada sem a emissão de um novo certificado. Identificado o problema, a solução exige contato com a AC emissora para revogação do certificado atual e emissão de um novo com algoritmo seguro.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- A escolha do **algoritmo de assinatura** não implica custo adicional em relação à emissão de um certificado convencional.
- Caso o certificado atual utilize **MD5 ou SHA-1**, será necessário **revogá-lo e emitir um novo**, o que pode envolver custo de reemissão dependendo da AC. Contudo, a maioria das ACs modernas já não emite certificados com esses algoritmos, e certificados **Let's Encrypt** não têm custo algum.
- Na prática, certificados com **MD5 ou SHA-1** são cada vez mais raros em emissões recentes, pois as principais ACs já **bloqueiam sua geração**.

🔧 Custo operacional estimado: Baixo

- A verificação do **algoritmo de assinatura** é imediata e pode ser feita gratuitamente por ferramentas como **SSL Labs** ou diretamente pelo navegador.
- A substituição do certificado segue o mesmo processo de uma **renovação convencional**, sem etapas adicionais complexas.
- Após a substituição, nenhuma manutenção adicional é necessária além do **ciclo normal de renovação** do certificado.

🛡️ Benefício estimado: Médio ou Alto

- Certificados assinados com **MD5 ou SHA-1** já são **bloqueados pelos principais navegadores**, o que na prática já força sua substituição independentemente deste controle.
- No entanto, ambientes com **navegadores desatualizados** ou clientes não convencionais podem ainda aceitar esses certificados, mantendo a vulnerabilidade ativa.
- A **falsificação de certificados** por **colisão de hash** é um ataque de alto impacto, pois compromete a identidade do servidor perante todos os usuários, justificando a substituição mesmo em cenários de risco aparentemente baixo.

3.14 Nome de Domínio no Certificado

O que é?

Um certificado digital é emitido para um domínio específico. O campo SAN (Subject Alternative Name) é o parâmetro do certificado que lista todos os domínios e subdomínios para os quais ele é válido.

Quando um usuário acessa um site, o navegador compara o endereço digitado com os domínios listados no SAN do certificado apresentado pelo servidor. Se houver divergência, o navegador exibe um alerta crítico de segurança e bloqueia o acesso.

💡 **Analogia:** é como um crachá de acesso corporativo emitido para um setor específico: não serve para acessar outro setor, mesmo sendo da mesma empresa. Um certificado emitido para www.dominio.gov.br não é válido para seguro.dominio.gov.br ou para dominio.gov.br (sem o “www”), a menos que esses endereços estejam explicitamente listados no campo SAN do certificado.

ℹ️ **Nota histórica:** antigamente, o domínio era especificado apenas no campo CN (Common Name). Atualmente, os navegadores modernos ignoram o CN e verificam

> exclusivamente o campo SAN. Certificados que não incluam o SAN são rejeitados pelos principais navegadores.

Como é verificado?

O teste verifica se o domínio do site corresponde a um dos domínios listados no campo SAN do certificado apresentado pelo servidor. O teste também avalia se variações comuns do domínio, como com e sem "www", estão cobertas.

Riscos do não atendimento

Risco	Consequência
Bloqueio pelo navegador	O navegador exibe um alerta crítico de segurança e impede o acesso ao site, comprometendo a disponibilidade para todos os usuários
Perda de confiança	Usuários que se deparam com alertas de certificado tendem a abandonar o site ou, pior, a desenvolver o hábito de ignorar alertas de segurança, tornando-os vulneráveis a ataques futuros
Ataque MitM	Um invasor pode explorar a divergência de domínio para apresentar um certificado de outro domínio, dificultando que usuários menos atentos percebam a irregularidade

Como resolver?

- **Emitir o certificado** com o campo **SAN** incluindo todos os domínios e subdomínios pelos quais o site será acessado
- Incluir tanto a versão com quanto sem **“www”** do domínio (ex.: dominio.gov.br e www.dominio.gov.br)
- Para sites com múltiplos subdomínios, considerar o uso de um certificado **Wildcard** (ex.: *.dominio.gov.br) ou **Multi-SAN**, que cobre múltiplos domínios em um único certificado

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — Subject Alternative Name	Documentação da Mozilla sobre o campo SAN, seu funcionamento e importância para a validação de domínios pelos navegadores	developer.mozilla.org/en-US/docs/Web/Security/Certificate_Transparency
CA/Browser Forum — Baseline Requirements	Requisitos técnicos mínimos que definem as regras para inclusão de domínios no campo SAN de certificados emitidos por ACs públicas	cabforum.org/working-groups/server/baseline-requirements
Let's Encrypt — Documentação	Guia para emissão de certificados com múltiplos domínios no campo SAN, incluindo certificados Wildcard e Multi-SAN	letsencrypt.org/docs/glossary/#def-SAN
RFC 5280	Especificação técnica do campo SAN no formato de certificados X.509, incluindo regras de correspondência de domínios	datatracker.ietf.org/doc/html/rfc5280#section-4.2.1.6
RFC 9525	Especificação das regras de verificação de identidade de servidores TLS, incluindo o uso obrigatório do campo SAN e a descontinuação do CN	datatracker.ietf.org/doc/html/rfc9525
SSL Labs — SSL Test	Verifica os domínios listados no campo SAN do certificado do servidor e identifica divergências com o domínio acessado	ssllabs.com/ssltest
ITI — ICP-Brasil	Requisitos técnicos para inclusão de domínios em certificados digitais no âmbito da ICP-Brasil	iti.gov.br

💡 **Dica:** o campo SAN do certificado pode ser verificado diretamente no navegador, clicando no cadeado na barra de endereços e acessando os detalhes do certificado. A seção "Nomes alternativos do titular" (ou "Subject Alternative Names") lista todos os domínios cobertos. O SSL Labs (ssllabs.com/ssltest) também exibe essa

informação na seção "Certificate" do relatório, facilitando a verificação de cobertura completa de todos os domínios e subdomínios utilizados.

⚠️ **Atenção para órgãos públicos:** certificados Wildcard (ex.: *.dominio.gov.br) cobrem apenas um nível de subdomínio — sub.dominio.gov.br é coberto, mas sub.sub.dominio.gov.br não é. Para estruturas com múltiplos níveis de subdomínio, é necessário listar cada um explicitamente no campo SAN ou emitir certificados específicos por nível. Verificar antecipadamente todos os endereços pelos quais o serviço será acessado antes de solicitar o certificado à AC evita a necessidade de reemissão.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- Certificados com SAN para múltiplos domínios ou subdomínios estão disponíveis **gratuitamente via Let's Encrypt**, sem custo adicional em relação a certificados de domínio único.
- Certificados **Wildcard** e **Multi-SAN** de ACs comerciais têm custo mais elevado, variando de algumas centenas a alguns milhares de reais por ano, dependendo da AC e do número de domínios cobertos.
- Na maioria dos casos, o custo financeiro é **nulo**, pois a correção resume-se a **reemitir o certificado** incluindo os domínios faltantes, processo geralmente gratuito nas principais ACs.

🔧 Custo operacional estimado: Baixo

- A correção resume-se à **reemissão do certificado** com os domínios corretos no campo SAN, seguida da reinstalação no servidor, processo idêntico a uma renovação convencional.
- Ferramentas como **SSL Labs** e a inspeção do certificado no próprio navegador permitem verificar imediatamente quais domínios estão cobertos.
- O principal ponto de atenção é realizar um **mapeamento prévio** de todos os domínios e subdomínios utilizados antes da emissão, evitando reemissões desnecessárias.

🛡️ Benefício estimado: Médio a Alto

- A **divergência de domínio** no certificado causa bloqueio imediato pelo navegador, tornando o problema evidente e urgente, o que na prática já força a correção independentemente deste controle.
- No entanto, a ausência de domínios no SAN pode passar **despercebida em subdomínios menos acessados**, mantendo vulnerabilidades silenciosas.
- A emissão correta do certificado desde o início, cobrindo todos os domínios necessários, tem **custo praticamente nulo** e elimina completamente essa classe de problemas.

3.15 Existência de DANE

O que é?

O DANE (DNS-Based Authentication of Named Entities) é uma técnica que permite vincular um certificado digital diretamente ao DNS do domínio, por meio de um registro especial chamado TLSA. Com isso, o cliente pode verificar a autenticidade do certificado não apenas pela cadeia de confiança tradicional das Autoridades Certificadoras (ACs), mas também pelo próprio DNS do domínio.

💡 **Analogia:** é como registrar a foto do seu documento de identidade em um banco de dados oficial e público. Se alguém apresentar um documento afirmando ser você, é possível consultar esse banco de dados para confirmar a veracidade, independentemente de quem o emitiu. O DANE atua de maneira idêntica: vincula o certificado do servidor diretamente ao DNS do domínio.

⚠️ **Pré-requisito obrigatório:** o DANE depende do DNSSEC — ver controle 1.1 (Existência de DNSSEC) — para funcionar. Sem DNSSEC, os registros `TLSA` no DNS podem ser falsificados, tornando o DANE ineficaz. Este teste terá resultado negativo caso o domínio não possua DNSSEC configurado corretamente.

Como é verificado?

O teste verifica se os servidores DNS do domínio contêm um registro `TLSA` válido, corretamente assinado pelo DNSSEC, e se ele corresponde ao certificado apresentado pelo servidor.

Riscos da ausência do DANE

Risco	Consequência
Comprometimento de AC	Se uma AC for invadida ou agir de má-fé, pode emitir certificados falsos para qualquer domínio. Sem DANE, navegadores aceitarão esses certificados como legítimos, viabilizando ataques MitM indetectáveis
Ataque MitM via certificado falso	Com um certificado falso emitido por uma AC comprometida, o invasor pode interceptar e adulterar toda a comunicação sem que o usuário perceba, pois o navegador não emitirá alertas

> ℹ️ **Caso real:** em 2011, a AC holandesa DigiNotar foi comprometida e emitiu certificados falsos para domínios de grandes organizações, incluindo o Google. Navegadores sem mecanismos adicionais de verificação, como o DANE, aceitaram esses certificados como legítimos, expondo milhares de usuários a ataques MitM.

Como resolver?

- Garantir que o **DNSSEC** esteja configurado corretamente no domínio — **pré-requisito**: ver controle 1.1 (Existência de DNSSEC)
- Publicar no DNS o **registro** `TLSA` correspondente ao certificado do servidor
- Manter o registro `TLSA` **sempre atualizado** a cada renovação de certificado, evitando divergências — a sequência correta de atualização está detalhada no controle 3.16 (Validade de DANE)

📚 Referências e documentação oficial

Referência	Descrição	URL
Registro.br — Ferramentas DNS	Ferramenta oficial do registro.br para verificação de registros `TLSA` e validação da configuração DANE em domínios .br	registro.br/tecnologia/ferramentas/geracao-de-registro-tlsa/
DNSViz	Ferramenta visual de análise e diagnóstico de DNSSEC e DANE, permitindo verificar a validade dos registros `TLSA` e sua integração com a cadeia DNSSEC	dnsviz.net
RFC 6698	Especificação técnica do protocolo DANE e do formato do registro `TLSA`	datatracker.ietf.org/doc/html/rfc6698
RFC 7671	Atualização da especificação DANE com clarificações e boas práticas para implementação do registro `TLSA`	datatracker.ietf.org/doc/html/rfc7671
NIST SP 800-81-3	Diretrizes do NIST para implantação segura de DNSSEC, base obrigatória para o funcionamento do DANE	csrc.nist.gov/pubs/sp/800/81/r3/final

💡 **Dica:** o DNSViz (dnsviz.net) permite verificar visualmente toda a cadeia de validação DANE, desde o registro `TLSA` no DNS até o certificado apresentado pelo servidor, identificando problemas de configuração antes que causem interrupção de serviço. É especialmente útil após renovações de certificado, quando o registro `TLSA` precisa ser atualizado para refletir o novo certificado.

⚠️ **Atenção crítica:** ao renovar um certificado, o registro `TLSA` deve ser atualizado *antes* que o novo certificado entre em produção. A sequência incorreta — trocar o certificado antes de atualizar o TLSA — causa falha de validação DANE para todos os clientes que verificam o registro, podendo tornar o serviço inacessível. A sequência

completa e recomendada de renovação está detalhada no controle 3.16 (Validade de DANE).

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- A publicação do **registro** `TLSA` é feita na zona DNS do domínio, sem custo financeiro direto além do já incorrido com o DNSSEC.
- O principal custo financeiro potencial é indireto: caso o provedor DNS atual **não suporte registros** `TLSA`, pode ser necessário migrar para um provedor compatível, com custos variando de gratuito a algumas centenas de reais por ano.

🔧 Custo operacional estimado: Médio

- A configuração inicial exige **conhecimento técnico específico** sobre DNSSEC e registros `TLSA`, além da correta geração do **hash do certificado** para o registro.
- O maior desafio operacional é a **manutenção contínua**: o registro `TLSA` precisa ser atualizado antes de cada renovação de certificado, pois uma divergência entre o certificado ativo e o registro `TLSA` pode tornar o site inacessível para clientes que validam o DANE.
- A recomendação de boas práticas é manter **dois registros** `TLSA` **simultâneos** durante a transição entre certificados, garantindo continuidade do serviço durante a renovação.

🛡️ Benefício estimado: Baixo

- O cenário de **comprometimento de uma AC**, embora documentado, é relativamente raro e geralmente detectado e corrigido rapidamente pelas próprias ACs e pelos fabricantes de navegadores, por meio de mecanismos como o **Certificate Transparency (CT)**.
- A adoção do DANE por **navegadores web ainda é limitada**: atualmente, os principais navegadores, como Chrome e Firefox, não validam registros `TLSA` para conexões HTTPS, reduzindo o impacto prático do controle no contexto web. O DANE é significativamente mais relevante e adotado no **contexto de e-mail**.
- Apesar do benefício baixo isoladamente, o DANE representa uma **camada adicional de segurança com custo financeiro nulo**, tornando sua implementação vantajosa em

domínios que já possuem DNSSEC configurado.

3.16 Validade de DANE

O que é?

Enquanto o controle 3.15 (Existência de DANE) verifica a existência do registro `TLSA` no DNS, este controle verifica se esse registro é válido, ou seja, se o hash publicado no registro `TLSA` corresponde efetivamente ao certificado atual do servidor.

> 💡 **Analogia:** ter o registro `TLSA` publicado no DNS é como exibir um crachá de identificação na entrada. Verificar a validade do DANE é confirmar que a foto no crachá realmente corresponde à pessoa que o apresenta. Um crachá existente, mas com foto errada, não oferece proteção real.

A situação mais comum de invalidade do DANE ocorre quando o certificado do servidor é renovado, mas o registro `TLSA` não é atualizado na mesma operação, gerando uma divergência entre o certificado ativo e o hash registrado no DNS.

Como é verificado?

O teste verifica se o hash contido no registro `TLSA` do domínio corresponde ao certificado atualmente apresentado pelo servidor web.

Riscos do não atendimento

Risco	Consequência
DANE inválido, proteção nula	Um registro `TLSA` desatualizado ou incorreto não oferece proteção real contra certificados falsos, anulando o benefício do controle 3.15 (Existência de DANE)

Risco	Consequência
Interrupção do serviço	Clientes que validam DANE, como servidores de e-mail, rejeitam conexões quando o registro `TLSA` não corresponde ao certificado apresentado, podendo causar indisponibilidade do serviço
Ataque MitM	Com o DANE inválido, o invasor que possua um certificado falso emitido por uma AC comprometida não encontra a barreira adicional que o DANE deveria oferecer

Como resolver?

- Atualizar o **registro** `TLSA` sempre que o certificado for renovado, preferencialmente **antes** da troca do certificado ativo
- Adotar o esquema de **dois registros** `TLSA` **simultâneos** durante a transição entre certificados:
  - **Antes da renovação:** publicar o registro `TLSA` do novo certificado junto ao do certificado atual
  - **Após a renovação:** remover o registro `TLSA` do certificado antigo

⚠️ **Atenção:** manter apenas um registro `TLSA` e atualizá-lo somente após a troca do certificado causa uma janela de invalidade, durante a qual o serviço pode ser interrompido para clientes que validam DANE.

Sequência recomendada para renovação de certificado com DANE

1. Gerar o novo certificado (ainda não ativo no servidor)
2. Calcular o hash TLSA do novo certificado
3. Publicar o novo registro `TLSA` no DNS
   mantendo o registro `TLSA` do certificado atual
4. Aguardar a propagação DNS (TTL do registro `TLSA`)
5. Substituir o certificado no servidor
6. Verificar a validade do DANE com DNSViz ou ferramenta equivalente
7. Remover o registro `TLSA` do certificado antigo

📚 Referências e documentação oficial

Referência	Descrição	URL
DNSViz	Ferramenta visual de diagnóstico DANE que permite verificar se o hash no registro `TLSA` corresponde ao certificado atual do servidor	dnsviz.net
RFC 6698	Especificação técnica do protocolo DANE, incluindo o formato do hash no registro `TLSA` e as regras de correspondência com o certificado	datatracker.ietf.org/doc/html/rfc6698
RFC 7671	Boas práticas para implementação do DANE, com orientações específicas sobre gestão do ciclo de vida do registro `TLSA` durante renovações de certificado	datatracker.ietf.org/doc/html/rfc7671
RFC 8555 — ACME	Especificação do protocolo ACME, utilizado pelo Let's Encrypt e outras ACs para automação da emissão e renovação de certificados, reduzindo o risco de desatualização do TLSA	datatracker.ietf.org/doc/html/rfc8555

Referência	Descrição	URL
Registro.br — Ferramentas DNS	Ferramenta oficial do registro.br para verificação de registros `TLSA` e validação da correspondência com o certificado do servidor	registro.br/tecnologia/ferramentas/geracao-de-registro-tlsa/

💡 **Dica:** o DNSViz (dnsviz.net) permite verificar visualmente se o hash no registro `TLSA` corresponde ao certificado atual do servidor, identificando divergências antes que causem interrupção de serviço. Recomenda-se executar essa verificação imediatamente após qualquer renovação de certificado ou alteração de zona DNS.

⚠️ **Atenção para automação:** sistemas que automatizam a renovação de certificados, como o Certbot com Let's Encrypt, não atualizam automaticamente o registro `TLSA` no DNS. É necessário integrar explicitamente a atualização do TLSA ao processo de renovação, seja por script personalizado ou por solução de automação DNS que suporte DANE. A ausência dessa integração é a causa mais comum de invalidade de DANE em ambientes que utilizam renovação automática de certificados.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A atualização do **registro** `TLSA` é feita na **zona DNS** do domínio, sem custo financeiro adicional além do já incorrido com o **DNSSEC** e o DANE.

🔧 Custo operacional estimado: Médio

- O custo operacional é médio porque a **manutenção da validade do DANE** exige sincronização entre dois processos distintos: a **renovação do certificado** e a **atualização do registro** `TLSA` **no DNS**, aumentando a complexidade operacional do ciclo de vida do certificado.
- O risco de **erro humano** é relevante: uma renovação de certificado realizada sem a correspondente atualização do TLSA invalida imediatamente o DANE e pode causar **interrupção do serviço**.

- A mitigação desse risco exige a adoção do esquema de **dois registros simultâneos** e, idealmente, a **automação** do processo de atualização do TLSA integrada ao processo de renovação do certificado, o que demanda esforço técnico adicional de configuração.

🛡️ Benefício estimado: Baixo

- O benefício é classificado como baixo pelas mesmas razões do controle 3.15 (Existência de DANE), com um agravante: um **registro** `TLSA` **existente, mas inválido**, oferece **proteção nula** e ainda pode causar interrupção de serviço, tornando a manutenção contínua da validade tão importante quanto a existência do registro.
- O benefício real do DANE, tanto em existência quanto em validade, é mais significativo no **contexto de e-mail** do que no contexto web, onde os principais navegadores ainda não validam registros `TLSA`.
- Apesar do benefício classificado como baixo, **manter o registro** `TLSA` **válido** é condição necessária para que o investimento operacional já realizado na implementação do DANE — ver controle 3.15 (Existência de DANE) — produza qualquer efeito prático de segurança.

4. Cabeçalhos de Segurança HTTP

Cabeçalhos de segurança HTTP são instruções enviadas pelo servidor ao navegador do usuário, ensinando-o a se defender de determinadas fraudes e ataques. São configurados no servidor web e não exigem nenhuma ação do usuário final.

> 💡 **Analogia:** funcionam como os regulamentos internos fixados na entrada de um edifício. Eles orientam todos os visitantes sobre as regras e restrições do espaço de forma automática, sem a necessidade de instruir cada pessoa individualmente.

4.1 X-Frame-Options

O que é?

O cabeçalho `X-Frame-Options` impede que a página do site seja exibida dentro de um "quadro" (frame ou iframe) embutido em outro site. Isso protege contra ataques de clickjacking, nos quais o invasor sobrepõe uma página legítima de forma invisível sobre outra página maliciosa, enganando o usuário para que clique em elementos sem perceber.

> 💡 **Exemplo de clickjacking:** o usuário acessa um site aparentemente inofensivo e clica no que parece ser um botão de "Jogar". Na verdade, por baixo, há um iframe invisível com o botão "Confirmar transferência" do seu banco. O clique é registrado no site legítimo, executando a ação sem o consentimento do usuário.

> ⚠️ **Nota de atualização:** o cabeçalho `Content-Security-Policy` — ver controle 4.3 (Content-Security-Policy) — por meio da diretiva frame-ancestors, torna o `X-Frame-Options` obsoleto em navegadores modernos. A recomendação atual é implementar ambos, garantindo compatibilidade com navegadores mais antigos.

Como é verificado?

O teste verifica se o servidor fornece o cabeçalho `X-Frame-Options` com uma política suficientemente segura, como DENY (bloqueia qualquer enquadramento) ou SAMEORIGIN

(permite enquadramento apenas pelo próprio domínio).

Riscos da ausência

Risco	Consequência
Clickjacking	O invasor emoldura a página legítima em um iframe invisível, induzindo o usuário a executar ações não intencionais, como confirmar transações ou alterar configurações
Roubo de informações	O usuário insere dados confidenciais, como senhas e números de cartão, em campos de uma página legítima enquadrada de forma maliciosa
Phishing por enquadramento	O invasor cria uma página falsa que enquadra a página legítima, capturando as interações do usuário

Como resolver?

- Adicionar o cabeçalho `X-Frame-Options` no servidor web com o valor mais restritivo compatível com o funcionamento do site
- Utilizar `DENY` quando a página não precisar ser enquadrada por nenhum domínio, incluindo o próprio
- Utilizar `SAMEORIGIN` quando a página precisar ser enquadrada pelo próprio domínio
- Implementar também a diretiva **frame-ancestors** no cabeçalho `Content-Security-Policy` — ver controle 4.3 (Content-Security-Policy) — que oferece controle mais granular e é prioritária em navegadores modernos

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla X-Frame-Options	Documentação oficial da Mozilla sobre o cabeçalho `X-Frame-Options`, com descrição dos valores aceitos e exemplos de configuração	developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options

Referência	Descrição	URL
OWASP — Clickjacking Defense Cheat Sheet	Guia prático da OWASP sobre proteção contra clickjacking, incluindo uso do `X-Frame-Options` e da diretiva CSP frame-ancestors	cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html
Mozilla Observatory	Ferramenta da Mozilla para análise de cabeçalhos de segurança HTTP, incluindo verificação do `X-Frame-Options`	observatory.mozilla.org
RFC 7034	Especificação técnica do cabeçalho `X-Frame-Options`	datatracker.ietf.org/doc/html/rfc7034
Security Headers	Ferramenta de análise de cabeçalhos de segurança HTTP, com verificação e avaliação do `X-Frame-Options` e demais cabeçalhos de segurança	securityheaders.com

💡 **Dica:** utilize o Mozilla Observatory (observatory.mozilla.org) ou o Security Headers (securityheaders.com) para verificar rapidamente se o cabeçalho `X-Frame-Options` está presente e corretamente configurado. Ambas as ferramentas analisam todos os cabeçalhos de segurança HTTP simultaneamente, permitindo uma visão consolidada da configuração do servidor — útil para verificar o conjunto de controles da seção 4 deste manual em uma única consulta.

⚠️ **Atenção:** sites que utilizam funcionalidades legítimas de enquadramento, como painéis integrados ou widgets incorporados em portais, devem avaliar cuidadosamente o valor utilizado. O uso de DENY bloqueia qualquer enquadramento, inclusive pelo próprio domínio. Nesses casos, SAMEORIGIN ou a diretiva frame-ancestors do CSP oferecem controle mais preciso sobre quais origens podem enquadrar a página.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Trata-se da adição de um **cabeçalho HTTP** na configuração do servidor web, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à adição de uma linha nos arquivos do servidor web. O único ponto de atenção é verificar se o site utiliza **iframes legítimos** de outros domínios, o que pode exigir ajuste na **política escolhida**.

🛡️ Benefício estimado: Alto

- Ataques de **clickjacking** são **silenciosos e difíceis de detectar** pelo usuário. O `X-Frame-Options` **elimina completamente** essa classe de ataques com custo praticamente nulo, justificando o benefício alto.

4.2 X-Content-Type-Options

O que é?

Os navegadores possuem um recurso chamado MIME sniffing, que tenta "adivinhar" o tipo de um arquivo quando o servidor não o informa claramente, ou quando o tipo declarado parece incorreto. O cabeçalho `X-Content-Type-Options` desativa esse comportamento, forçando o navegador a respeitar o tipo de conteúdo declarado pelo servidor.

> 💡 **Analogia:** é como tentar adivinhar o conteúdo de uma caixa selada examinando sua aparência, mesmo que a etiqueta já informe o que está dentro. O `X-Content-Type-Options` instrui o navegador a confiar apenas na etiqueta declarada e nunca “adivinhar” por conta própria — impedindo que um arquivo rotulado como imagem seja identificado como script e executado.

Como é verificado?

O teste verifica se o servidor fornece o cabeçalho `X-Content-Type-Options: nosniff`.

Riscos da ausência

Risco	Consequência
MIME Sniffing	O navegador ignora o tipo de conteúdo declarado pelo servidor e executa um arquivo malicioso como script
Cross-Site Scripting (XSS) via upload	Um invasor faz upload de um arquivo com conteúdo malicioso que o navegador interpreta e executa como JavaScript

Como resolver?

- Adicionar o cabeçalho `X-Content-Type-Options` com o valor `nosniff` no servidor web — este é o único valor definido pela especificação e instrui o navegador a bloquear qualquer requisição cujo tipo de conteúdo não corresponda ao declarado pelo servidor
- Garantir que o servidor declare corretamente o **Content-Type** de todos os recursos servidos, pois o `nosniff` depende de tipos declarados corretos para funcionar adequadamente

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla X-Content-Type-Options	Documentação oficial da Mozilla sobre o cabeçalho `X-Content-Type-Options`, com descrição do comportamento do nosniff e casos de uso	developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options
Mozilla — MIME types	Documentação da Mozilla sobre tipos MIME e os riscos do MIME sniffing em navegadores	developer.mozilla.org/en-US/docs/Web/HTTP/MIME_types

Referência	Descrição	URL
OWASP — MIME Sniffing	Documentação da OWASP sobre o risco de MIME sniffing e as medidas de mitigação recomendadas	owasp.org/www-project-secure-headers
Mozilla Observatory	Ferramenta da Mozilla para análise de cabeçalhos de segurança HTTP, incluindo verificação do `X-Content-Type-Options`	observatory.mozilla.org
Security Headers	Ferramenta de análise de cabeçalhos de segurança HTTP, com verificação e avaliação do `X-Content-Type-Options` e demais cabeçalhos de segurança	securityheaders.com
WHATWG — Fetch Standard	Especificação técnica que define o comportamento do MIME sniffing e o efeito do cabeçalho nosniff nos navegadores modernos	fetch.spec.whatwg.org

💡 **Dica:** o valor nosniff é o único definido pela especificação para este cabeçalho — qualquer outro valor é ignorado pelos navegadores. A configuração é simples e sem efeitos colaterais em servidores que já declaram corretamente o Content-Type de seus recursos, o que torna este um dos controles de menor risco de implementação deste manual.

⚠️ **Atenção:** o nosniff pode causar bloqueios em servidores que declaram tipos de conteúdo incorretos ou genéricos, como application/octet-stream para recursos que deveriam ter tipos específicos. Antes de habilitar o cabeçalho, verificar se todos os recursos do site estão sendo servidos com os tipos MIME corretos para evitar quebras de funcionalidade.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Adição de um **cabeçalho HTTP** na configuração do servidor web, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- Configuração de uma única linha no servidor web, sem impacto esperado em funcionalidades legítimas quando os **tipos MIME** estão corretamente declarados.

🛡️ Benefício estimado: Alto

- **Elimina uma classe de ataques** que explora um comportamento padrão dos navegadores, com **custo zero de implementação**, justificando o benefício alto.

4.3 Content-Security-Policy (CSP)

O que é?

O `Content-Security-Policy` (CSP) é o cabeçalho de segurança mais abrangente disponível. Ele permite que o administrador do site defina uma lista de origens confiáveis para carregar scripts, estilos, imagens e outros recursos, impedindo que o navegador execute conteúdo proveniente de fontes não autorizadas.

> 💡 **Analogia:** o CSP funciona como uma lista de fornecedores aprovados. O navegador só aceita conteúdo de origens que estejam explicitamente na lista, bloqueando qualquer recurso de origem desconhecida, mesmo que esteja embutido na própria página.

ℹ️ **Relação com o controle 4.1:** a diretiva **frame-ancestors** do CSP substitui o `X-Frame-Options` — ver controle 4.1 (X-Frame-Options) — em navegadores modernos. Recomenda-se implementar ambos para garantir compatibilidade ampla.

Como é verificado?

O teste verifica se o servidor fornece o cabeçalho `Content-Security-Policy` com uma política definida.

Riscos da ausência

Risco	Consequência
Cross-Site Scripting (XSS)	Scripts maliciosos injetados por terceiros são executados no navegador do usuário, podendo roubar credenciais, sequestrar sessões e distribuir malware
Clickjacking	Sem a diretiva frame-ancestors, a página pode ser enquadrada em sites maliciosos
Injeção de conteúdo	Recursos de origens não autorizadas, como scripts de rastreamento ou mineradores de criptomoedas, são carregados e executados sem o conhecimento do administrador

Como resolver?

Adicionar o cabeçalho `Content-Security-Policy` no servidor web com uma política restritiva, definindo explicitamente as origens confiáveis para cada tipo de recurso. O CSP deve ser implantado de forma gradual, em **três etapas**, para evitar bloqueios não intencionais de recursos legítimos:

Etapa	Abordagem	Objetivo
1. Monitoramento	Utilizar `Content-Security-Policy-Report-Only` com a política desejada	Identificar recursos bloqueados sem afetar o funcionamento do site

Etapa	Abordagem	Objetivo
2. Refinamento	Analisar os relatórios de violação e ajustar a política	Garantir que todos os recursos legítimos estejam cobertos
3. Aplicação	Substituir Report-Only pelo cabeçalho `Content-Security-Policy` definitivo	Aplicar a política com bloqueio efetivo

> ⚠️ **Atenção:** uma política CSP mal configurada pode bloquear recursos legítimos do site, como scripts de analytics, fontes externas ou sistemas de autenticação de terceiros, causando quebra de funcionalidades. O uso do modo Report-Only na etapa inicial é essencial para evitar esse problema.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla — `Content-Security-Policy`	Documentação oficial da Mozilla sobre o cabeçalho CSP, com descrição completa de todas as diretivas disponíveis e exemplos de configuração	developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy
OWASP — Content Security Policy Cheat Sheet	Guia prático da OWASP sobre implementação do CSP, incluindo estratégias de implantação gradual e boas práticas	cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html
CSP Evaluator	Ferramenta do Google para análise e avaliação de políticas CSP, identificando diretivas ausentes, configurações inseguras e sugestões de melhoria	csp-evaluator.withgoogle.com
Mozilla Observatory	Ferramenta da Mozilla para análise de cabeçalhos de segurança HTTP, com avaliação detalhada da política CSP configurada	observatory.mozilla.org

Referência	Descrição	URL
Security Headers	Ferramenta de análise de cabeçalhos de segurança HTTP, com verificação e avaliação do CSP e demais cabeçalhos de segurança	securityheaders.com
W3C — CSP Level 3	Especificação técnica oficial do Content Security Policy nível 3, com definição completa de todas as diretivas e seu comportamento	w3.org/TR/CSP3

💡 **Dica:** o CSP Evaluator (csp-evaluator.withgoogle.com) é a ferramenta mais prática para validar uma política CSP antes de aplicá-la em produção. Ele identifica diretivas ausentes, valores inseguros como `unsafe-inline` e `unsafe-eval`, e apresenta sugestões específicas de melhoria. Utilize-o em conjunto com o modo **Report-Only** durante a fase de refinamento da política.

⚠️ **Atenção para sistemas com conteúdo dinâmico:** sites que utilizam CMS, frameworks JavaScript ou integrações com serviços de terceiros tendem a ter políticas CSP mais complexas, pois precisam listar todas as origens externas utilizadas. Nesses casos, o período de monitoramento com Report-Only é especialmente importante para mapear todas as origens antes de aplicar a política restritiva.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Adição de um **cabeçalho HTTP** na configuração do servidor web, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo a Médio

- Embora a adição do cabeçalho seja simples, a definição de uma **política CSP eficaz** pode exigir esforço considerável em sites complexos, com múltiplas origens de conteúdo legítimas (ex.: **CDNs**, serviços de análise, fontes externas). O uso do modo **Report-Only**

antes da aplicação definitiva reduz o risco de quebra de funcionalidades, mas aumenta o tempo de implementação.

🛡️ Benefício estimado: Alto

- O CSP é uma das defesas mais eficazes contra **XSS**, um dos ataques mais comuns e impactantes em aplicações web. Quando bem configurado, **elimina ou mitiga significativamente** toda uma classe de **ataques de injeção de código**, justificando o benefício alto.

4.4 Referrer-Policy

O que é?

Quando um usuário clica em um link, o navegador envia ao site de destino o endereço da página de origem, por meio do cabeçalho 'Referer'. Esse endereço pode conter informações sensíveis, como tokens de autenticação, termos de pesquisa ou identificadores de sessão presentes na URL. O cabeçalho `Referrer-Policy` controla quanta informação da URL de origem é compartilhada com o site de destino.

> 💡 **Analogia:** é como usar um envelope com remetente. Ao encaminhar uma carta a um destinatário externo, você pode optar por incluir o endereço completo da origem, apenas o nome da empresa, ou não informar nada. O `Referrer-Policy` faz exatamente isso com a URL: ao navegar para outro site, define quanta informação sobre a página de origem é compartilhada com o destino. Um endereço com parâmetros sensíveis, como `sistema.gov.br/painel?token=abc123`, pode vazar dados críticos se o navegador revelar a URL completa.

Como é verificado?

O teste verifica se o servidor fornece o cabeçalho `Referrer-Policy` com uma política definida.

Riscos da ausência

Risco	Consequência
Vazamento de tokens	Tokens de autenticação ou sessão presentes na URL são enviados a sites externos, podendo ser capturados e reutilizados
Exposição de dados sensíveis	Termos de pesquisa, identificadores de usuário e estrutura interna de URLs são revelados a terceiros
Facilitação de ataques	Informações sobre a estrutura do sistema expostas via Referer podem auxiliar um invasor no mapeamento do ambiente

Como resolver?

Adicionar o cabeçalho `Referrer-Policy` no servidor web com uma política compatível com o funcionamento do site. Os principais valores disponíveis são:

Valor	Comportamento	Recomendação
no-referrer	Nunca envia o cabeçalho Referer	✅ **Mais restritivo**
strict-origin-when-cross-origin	Envia apenas o domínio para requisições externas; URL completa para requisições internas	✅ **Recomendado para a maioria dos casos**
same-origin	Envia a URL completa apenas para requisições do mesmo domínio; nada para externas	✅ **Adequado para sistemas internos**
no-referrer-when-downgrade	Envia a URL completa para HTTPS→HTTPS; nada para HTTPS→HTTP	⚠️ **Antigo padrão dos navegadores, sem proteção adicional**
unsafe-url	Sempre envia a URL completa, independentemente da origem	❌ **Nunca utilizar**

> 💡 **Boa prática:** a política **strict-origin-when-cross-origin** é a recomendada para serviços públicos, pois equilibra privacidade e compatibilidade com sistemas de analytics e integrações externas legítimas.

📚 Referências e documentação oficial

Referência	Descrição	URL
Mozilla Referrer-Policy	Documentação oficial da Mozilla sobre o cabeçalho `Referrer-Policy`, com descrição completa de todos os valores disponíveis e seus comportamentos	developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Referrer-Policy
Mozilla — Referer header privacy and security concerns	Documentação da Mozilla sobre os riscos de privacidade e segurança associados ao cabeçalho Referer e como mitigá-los	developer.mozilla.org/en-US/docs/Web/Security/Referer_header:_privacy_and_security_concerns
OWASP — Secure Headers Project	Documentação da OWASP sobre cabeçalhos de segurança HTTP, incluindo recomendações para o `Referrer-Policy`	owasp.org/www-project-secure-headers
Mozilla Observatory	Ferramenta da Mozilla para análise de cabeçalhos de segurança HTTP, incluindo verificação do `Referrer-Policy`	observatory.mozilla.org
Security Headers	Ferramenta de análise de cabeçalhos de segurança HTTP, com verificação e avaliação do `Referrer-Policy` e demais cabeçalhos	securityheaders.com
W3C — Referrer Policy	Especificação técnica oficial do cabeçalho `Referrer-Policy`, com definição completa de todos os valores e seu comportamento nos navegadores	w3.org/TR/referrer-policy

💡 **Dica:** utilize o Mozilla Observatory (observatory.mozilla.org) ou o Security Headers (securityheaders.com) para verificar rapidamente se o cabeçalho `Referrer-Policy` está presente e configurado corretamente. A política **strict-origin-when-cross-origin** é atualmente o valor padrão dos navegadores modernos quando nenhuma política é definida explicitamente — mas declará-la explicitamente no servidor garante o comportamento esperado independentemente da versão do navegador do usuário.

⚠️ **Atenção:** sistemas que dependem do valor completo do Referer para funcionalidades legítimas, como controles de acesso baseados em origem ou sistemas de analytics, podem ser afetados por políticas mais restritivas. Nesses casos, avaliar se a funcionalidade pode ser implementada por meios alternativos que não dependam da URL completa de origem antes de aplicar uma política mais restritiva.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Adição de um **cabeçalho HTTP** na configuração do servidor web, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se à adição de uma linha no servidor web. O único ponto de atenção é verificar se alguma funcionalidade legítima depende do **Referer completo** para operar, o que pode exigir ajuste na **política escolhida**.

🛡️ Benefício estimado: Alto

- O **vazamento de informações sensíveis** via Referer é um **risco silencioso** e frequentemente ignorado. A `Referrer-Policy` elimina esse risco com **custo zero**, protegendo diretamente a privacidade e a segurança dos usuários, justificando o benefício alto.

5. Autenticação de E-mail

A autenticação de e-mail é composta por uma trilogia de protocolos complementares — SPF, DKIM e DMARC — que trabalham juntos para impedir que invasores enviem mensagens fingindo ser a sua organização. Nenhum deles, isoladamente, oferece proteção completa; a eficácia máxima é alcançada quando os três estão configurados corretamente.

> 💡 **Analogia:** o SPF é a lista de funcionários autorizados a falar em nome da organização. O DKIM é o crachá assinado que prova que quem fala é realmente quem diz ser. O DMARC é o regulamento que define o que fazer quando alguém se apresenta sem autorização ou com crachá falso.

5.1 Existência de DMARC

O que é?

O DMARC (Domain-Based Message Authentication, Reporting, and Conformance) é o protocolo que instrui os servidores de e-mail destinatários sobre o que fazer quando um e-mail falha nas verificações de SPF ou DKIM. Além disso, gera relatórios detalhados sobre tentativas de uso indevido do domínio, permitindo que o administrador monitore e corrija problemas de configuração.

> 💡 **Analogia:** o DMARC funciona como o regulamento da portaria de um edifício, que orienta o segurança sobre como agir quando um visitante chega sem crachá: permitir a entrada, encaminhar para triagem ou barrar. Sem esse regulamento, cada servidor destinatário age de maneira independente — alguns entregam a mensagem, enquanto outros a descartam.

> ℹ️ **Relação com SPF e DKIM:** sem o DMARC, mesmo que **SPF** e **DKIM** estejam configurados, os servidores destinatários não recebem nenhuma instrução clara sobre como tratar e-mails que falham nessas verificações, podendo entregá-los normalmente na caixa de entrada do destinatário.

Como é verificado?

O teste verifica se existe um registro DMARC publicado no DNS do domínio, no formato de um registro `TXT` no subdomínio _dmarc.dominio.gov.br.

Riscos da ausência

Risco	Consequência
Domain Spoofing	Criminosos enviam e-mails usando o domínio da organização como remetente, e os servidores destinatários não têm instrução para bloqueá-los
Phishing	Usuários recebem e-mails aparentemente legítimos do domínio da organização com conteúdo malicioso, sem que haja mecanismo de bloqueio
Ausência de visibilidade	Sem os relatórios do DMARC, o administrador não tem como saber que seu domínio está sendo usado indevidamente

Como resolver?

Publicar um registro **DMARC** no DNS do domínio. O **DMARC** deve ser implantado de forma **gradual**, em **três etapas**, para evitar bloqueios não intencionais de e-mails legítimos:

Etapa	Política	Objetivo
1. Monitoramento	`p=none`	Receber relatórios sem bloquear nenhum e-mail. Permite mapear todos os remetentes legítimos do domínio
2. Quarentena	`p=quarantine`	E-mails que falham nas verificações são movidos para a pasta de spam. Permite identificar falsos positivos antes do bloqueio total
3. Rejeição	`p=reject`	E-mails não autenticados são rejeitados pelo servidor destinatário. Proteção máxima contra spoofing

⚠️ **Atenção:** avançar diretamente para `p=reject` sem passar pelas etapas anteriores pode bloquear e-mails legítimos enviados por sistemas ainda não configurados com SPF e DKIM corretos, como plataformas de disparo de e-mail marketing, sistemas de helpdesk ou integrações com terceiros.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail, cobrindo SPF, DKIM e DMARC, com explicações em português e orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth/
Google — DMARC	Guia prático do Google para implementação do DMARC, com instruções detalhadas para cada etapa de implantação e análise de relatórios	support.google.com/a/answer/2466580
DMARC.org	Documentação oficial do protocolo DMARC, mantida pela organização responsável pela especificação	dmarc.org/overview
RFC 7489	Especificação técnica do protocolo DMARC	datatracker.ietf.org/doc/html/rfc7489
MXToolbox — DMARC Lookup	Ferramenta para verificação do registro DMARC publicado no DNS do domínio, com análise da política configurada	mxtoolbox.com/dmarc.aspx

💡 **Dica:** durante a etapa de monitoramento (`p=none`), configure o endereço de recebimento de relatórios DMARC (rua=mailto:dmarc@dominio.gov.br) e analise os relatórios recebidos por pelo menos 30 dias antes de avançar para a próxima etapa. Os relatórios revelam todos os sistemas que enviam e-mails em nome do domínio — informação essencial para evitar bloqueios não intencionais nas etapas seguintes.

⚠️ **Atenção para subdomínios:** por padrão, a política DMARC configurada no domínio principal (dominio.gov.br) se aplica automaticamente a todos os subdomínios (servico.dominio.gov.br, noreply.dominio.gov.br, etc.), a menos que cada subdomínio tenha seu próprio registro DMARC. Verificar se há subdomínios que enviam e-mails e que precisam de configuração específica antes de avançar para políticas mais restritivas.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A publicação do **registro DMARC** é feita na **zona DNS** do domínio, sem custo financeiro envolvido. **Ferramentas gratuitas** de monitoramento de relatórios DMARC estão disponíveis, embora **soluções pagas** ofereçam maior facilidade de análise.

🔧 Custo operacional estimado: Baixo

- A criação do **registro** é simples e bem documentada. O principal esforço está na análise dos **relatórios** gerados pelo **DMARC**, que exige atenção periódica para identificar e corrigir problemas de configuração de **SPF e DKIM** antes de aplicar políticas mais restritivas.

🛡️ Benefício estimado: Alto

- O **DMARC** é a única camada que efetivamente instrui outros servidores a bloquear ativamente e-mails falsificados com o domínio da organização. Sem ele, **SPF e DKIM** sozinhos não impedem a entrega de e-mails fraudulentos, tornando este controle essencial para a proteção da **reputação do domínio** e dos destinatários.

5.2 Política de DMARC

O que é?

A existência do registro DMARC não é suficiente por si só: a política definida nele determina o nível real de proteção oferecido. Existem três opções:

> 💡 **Analogia:** pense nos três níveis como um sistema de controle de acesso. Com `p=none`, a câmera registra quem entra sem autorização, mas ninguém é barrado. Com `p=quarantine`, os suspeitos são encaminhados para uma sala de espera (pasta de spam). Com `p=reject`, o segurança barra o acesso na porta. Ter DMARC com `p=none` é como ter câmera sem segurança: você vê o que acontece, mas não impede nada.

Política	O que instrui o servidor destinatário a fazer
`p=none`	Não toma nenhuma ação. Apenas coleta relatórios. Útil durante a fase de monitoramento inicial
`p=quarantine`	Direciona e-mails suspeitos para a pasta de spam
`p=reject`	Rejeita completamente e-mails suspeitos, impedindo sua entrega

> ⚠️ **Atenção:** uma política `p=none` equivale à ausência de proteção ativa. O DMARC existe, mas não bloqueia nenhum e-mail fraudulento.

Como é verificado?

O teste verifica se a política DMARC configurada é suficientemente rigorosa, ou seja, 'p=quarantine' ou 'p=reject'.

Riscos do não atendimento

Com 'p=none', e-mails fraudulentos que falsificam o domínio da organização continuam sendo entregues normalmente na caixa de entrada dos destinatários, tornando o DMARC ineficaz como mecanismo de proteção.

Como resolver?

A migração para uma **política ativa** deve ser feita de forma **gradual** e controlada. Avançar diretamente para `p=reject` sem passar pelas fases intermediárias pode bloquear e-mails legítimos enviados por sistemas ainda não cobertos pelo **SPF e DKIM**, como **plataformas de disparo**, sistemas de helpdesk ou integrações com terceiros. A sequência recomendada é:

Fase	Política	Condição para avançar
1. Monitoramento	`p=none`	Analisar relatórios por pelo menos 30 dias e identificar todos os remetentes legítimos do domínio
2. Endurecimento gradual	`p=quarantine`	Confirmar que todos os remetentes legítimos estão cobertos por SPF e DKIM válidos
3. Proteção máxima	`p=reject`	Ter confiança de que nenhum e-mail legítimo será rejeitado

⚠️ **Atenção:** não avançar para `p=reject` sem passar pelas fases anteriores. E-mails legítimos enviados por sistemas sem SPF e DKIM corretamente configurados — como plataformas de disparo, sistemas de helpdesk ou integrações com terceiros — serão rejeitados, causando falhas de comunicação.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail, cobrindo SPF, DKIM e DMARC, com explicações em português e orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth

Referência	Descrição	URL
Google — DMARC	Guia prático do Google para implementação gradual do DMARC, com instruções detalhadas para cada fase de implantação e análise de relatórios	support.google.com/a/answer/2466580
DMARC.org	Documentação oficial do protocolo DMARC, com orientações sobre políticas e boas práticas de implantação	dmarc.org/overview
RFC 7489	Especificação técnica do protocolo DMARC, incluindo definição formal das políticas none, quarantine e reject	datatracker.ietf.org/doc/html/rfc7489
MXToolbox — DMARC Lookup	Ferramenta para verificação da política DMARC publicada no DNS do domínio, com análise detalhada do registro configurado	mxtoolbox.com/dmarc.aspx

💡 **Dica:** ferramentas como o MXToolbox (mxtoolbox.com/dmarc.aspx) permitem verificar rapidamente a política DMARC atual do domínio e identificar se ela está em nível de proteção ativo. Durante a fase de monitoramento, utilize plataformas de análise de relatórios DMARC, como DMARC Analyzer ou Postmark DMARC, para interpretar os relatórios recebidos e identificar todos os remetentes legítimos antes de avançar para políticas mais restritivas.

⚠️ **Atenção para subdomínios:** a tag `sp` do DMARC permite definir uma política específica para subdomínios, independentemente da política do domínio principal. Caso existam subdomínios que não enviam e-mails, recomenda-se configurar `sp=reject` para bloquear qualquer tentativa de spoofing por meio deles, mesmo que o domínio principal ainda esteja em fase de transição.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A alteração da política é feita editando o **registro DMARC** já existente no **DNS**, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- O principal esforço é garantir que todos os **remetentes legítimos** do domínio, como sistemas de envio de **newsletters**, **alertas automáticos** e **parceiros**, estejam devidamente autorizados no **SPF** e assinando com **DKIM** antes de endurecer a política, para evitar bloqueio de e-mails legítimos.

🛡️ Benefício estimado: Alto

- A política `p=reject` é o estado final desejado e representa a **proteção máxima** contra **falsificação do domínio**. A diferença entre ter DMARC com `p=none` e com `p=reject` é a diferença entre monitorar ataques e bloqueá-los ativamente.

5.3 Existência de DKIM

O que é?

O DKIM (DomainKeys Identified Mail) é uma assinatura criptográfica adicionada ao cabeçalho de cada e-mail enviado. Essa assinatura permite que o servidor destinatário verifique que:

- O e-mail foi realmente enviado pelo domínio declarado
- O conteúdo não foi alterado durante o trajeto

O mecanismo funciona com um par de chaves: a chave privada assina o e-mail no servidor de envio, e a chave pública é publicada no DNS do domínio para que qualquer servidor possa verificar a assinatura.

> 💡 **Analogia:** o DKIM é como um lacre holográfico em uma correspondência oficial. Qualquer um pode verificar se o lacre é legítimo consultando o padrão publicado pelo órgão emissor, e qualquer adulteração no conteúdo rompe o lacre.

Como é verificado?

O teste verifica se o domínio possui registros DKIM publicados no DNS e se as assinaturas são válidas.

Riscos da ausência

Risco	Consequência
Falsificação de remetente	Sem DKIM, não há como provar criptograficamente que o e-mail foi enviado pelo domínio legítimo
Adulteração de conteúdo	E-mails podem ser modificados em trânsito sem que o destinatário perceba
Classificação como spam	Servidores destinatários modernos tendem a classificar e-mails sem DKIM como suspeitos, comprometendo a entregabilidade de e-mails legítimos

Como resolver?

1. Gerar um **par de chaves DKIM** no servidor de e-mail
2. Publicar a **chave pública** no **DNS** do domínio como **registro** `TXT`
3. Configurar o servidor de e-mail para assinar automaticamente todos os e-mails enviados com a **chave privada**

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail, cobrindo SPF, DKIM e DMARC, com explicações em português e orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth
Google — DKIM	Guia prático do Google para geração e publicação de chaves DKIM, com instruções detalhadas para os principais servidores de e-mail	support.google.com/a/answer/174124
RFC 6376	Especificação técnica do protocolo DKIM, incluindo o formato da assinatura e o mecanismo de verificação via DNS	datatracker.ietf.org/doc/html/rfc6376
MXToolbox — DKIM Lookup	Ferramenta para verificação do registro DKIM publicado no DNS do domínio, com análise da chave pública configurada	mxtoolbox.com/dkim.aspx

💡 **Dica:** domínios que utilizam múltiplos serviços de envio de e-mail — como servidor próprio, plataforma de e-mail marketing e sistema de helpdesk — precisam publicar um **registro DKIM** para cada serviço, pois cada um assina com sua própria **chave privada**. O seletor DKIM (`selector._domainkey.dominio.gov.br`) permite identificar qual chave corresponde a cada serviço. Verifique junto a cada fornecedor as instruções específicas para publicação do registro DKIM correspondente.

⚠️ **Atenção ao tamanho da chave:** chaves DKIM de 1024 bits são consideradas insuficientes pelos padrões atuais. Recomenda-se utilizar chaves de no mínimo 2048 bits, conforme orientação do RFC 8301. Ao gerar ou renovar chaves DKIM, verificar o tamanho configurado no servidor de e-mail e nos registros DNS publicados.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A geração de chaves e publicação no DNS não envolve custo financeiro. Em serviços de e-mail em nuvem, como **Google Workspace** e **Microsoft 365**, o **DKIM** é configurado gratuitamente na interface administrativa.

🔧 Custo operacional estimado: Baixo

- A configuração é bem documentada para os principais servidores e serviços de e-mail. O principal ponto de atenção é a **rotação periódica das chaves DKIM**, recomendada a cada **6 a 12 meses**, para limitar o impacto de um eventual comprometimento da **chave privada**.

🛡️ Benefício estimado: Alto

- O **DKIM** é o único mecanismo que garante a **integridade do conteúdo** do e-mail, complementando o **SPF** (que verifica o servidor de envio) e o **DMARC** (que define a política). Sua ausência compromete a confiabilidade de toda a cadeia de autenticação de e-mail.

5.4 Existência de SPF

O que é?

O SPF (Sender Policy Framework) é uma lista publicada no DNS do domínio que declara quais servidores e endereços IP estão autorizados a enviar e-mails em nome daquele domínio. Quando um e-mail chega ao servidor destinatário, ele consulta o SPF do domínio do remetente e verifica se o IP de origem está na lista autorizada.

> 💡 **Analogia:** o SPF é como uma lista oficial de veículos autorizados a fazer entregas em nome de uma organização. Se um veículo não está na lista, a entrega é

suspeita.

Como é verificado?

O teste verifica se existe um registro SPF publicado no DNS do domínio, como um registro `TXT` na zona do domínio.

Riscos da ausência

Risco	Consequência
Spoofing	Qualquer servidor pode enviar e-mails usando o domínio da organização como remetente, sem possibilidade de verificação
Phishing	E-mails fraudulentos em nome da organização são entregues sem obstáculos, aumentando a taxa de sucesso de ataques
Entregabilidade comprometida	E-mails legítimos enviados pelo domínio podem ser classificados como spam por servidores que exigem SPF

Como resolver?

Publicar um **registro SPF** no **DNS** do domínio como um **registro** `TXT`, listando todos os **servidores e endereços IP autorizados** a enviar e-mails em nome do domínio.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail, cobrindo SPF, DKIM e DMARC, com explicações em português e orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth
Google — SPF	Guia prático do Google para criação e publicação do registro SPF, com instruções detalhadas e exemplos de configuração	support.google.com/a/answer/33786

Referência	Descrição	URL
RFC 7208	Especificação técnica do protocolo SPF, incluindo o formato do registro, os mecanismos de verificação e as qualificações de resultado	datatracker.ietf.org/doc/html/rfc7208
MXToolbox — SPF Lookup	Ferramenta para verificação do registro SPF publicado no DNS do domínio, com análise dos mecanismos configurados e identificação de problemas	mxtoolbox.com/spf.aspx
SPF Record Checker	Ferramenta de verificação e validação de registros SPF, com análise detalhada dos mecanismos e identificação de erros de configuração	dmarcian.com/spf-survey

💡 **Dica:** domínios que utilizam múltiplos serviços de envio de e-mail precisam incluir todos os servidores autorizados no mesmo registro SPF. O SPF possui um limite de 10 consultas DNS por verificação — incluir muitos mecanismos `include:` pode ultrapassar esse limite e causar falhas de verificação. Ferramentas como o MXToolbox (mxtoolbox.com/spf.aspx) identificam quando esse limite está próximo de ser atingido.

⚠️ **Atenção:** assim como o **DMARC**, o **SPF** isoladamente não é suficiente para proteção completa. Um servidor destinatário pode verificar o SPF e constatar que o IP não está autorizado, mas sem uma política DMARC ativa (`p=quarantine` ou `p=reject`), nenhuma ação de bloqueio será tomada. A eficácia máxima é alcançada com **SPF**, **DKIM** e **DMARC** configurados corretamente em conjunto.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A publicação do **registro SPF** é feita na **zona DNS** do domínio, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- A criação do registro é simples. O principal desafio é mapear todos os **servidores e serviços** que enviam e-mails em nome do domínio (ex.: sistemas de notificação, **newsletters**, **CRMs**) antes de publicar o registro, para evitar que e-mails legítimos sejam bloqueados.

🛡️ Benefício estimado: Alto

- O **SPF** é a primeira linha de defesa contra **spoofing de domínio**. Sua ausência facilita diretamente ataques de **phishing** em nome da organização, com impacto direto sobre os cidadãos e a reputação institucional.

5.5 Política de SPF

O que é?

Assim como o DMARC, a simples existência do registro SPF não é suficiente: a política de falha define o que o servidor destinatário deve fazer quando um e-mail é enviado por um IP não autorizado.

> 💡 **Analogia:** imagine o SPF como um porteiro com uma lista de convidados. A política `~all` (softfail) equivale ao porteiro que anota o nome do visitante não convidado e permite sua entrada com um crachá de alerta — ele é considerado suspeito, mas não barrado. Já a política `-all` (hardfail) é o porteiro que barra a entrada imediatamente. Ter o SPF sem uma política de falha restritiva é como entregar a lista ao porteiro, mas não lhe dar instruções sobre como agir com quem está fora dela.

Política	Sintaxe	O que instrui o servidor a fazer
Hardfail	`-all`	Rejeitar definitivamente o e-mail
Softfail	`~all`	Marcar o e-mail como suspeito (geralmente vai para spam)

Política	Sintaxe	O que instrui o servidor a fazer
Neutral	`?all`	Não tomar nenhuma ação (equivale à ausência de política)

Como é verificado?

O teste verifica se o registro SPF contém uma política de falha suficientemente rigorosa: `~all` (softfail) ou `-all` (hardfail).

Riscos do não atendimento

Uma política neutra `?all` ou a ausência de política permite que e-mails enviados por IPs não autorizados sejam entregues normalmente, tornando o SPF ineficaz como mecanismo de proteção.

Como resolver?

A migração para uma **política mais restritiva** deve ser feita de forma **gradual**, após confirmar que todos os **remetentes legítimos** estão corretamente listados no **registro SPF**:

Fase	Política	Condição para avançar
1. Validação	`~all` (softfail)	Monitorar por pelo menos 30 dias e confirmar que nenhum remetente legítimo está sendo marcado como suspeito
2. Proteção máxima	`-all` (hardfail)	Ter confiança de que todos os servidores autorizados estão listados no registro SPF

⚠️ **Atenção:** avançar para -all sem validar todos os remetentes legítimos pode causar rejeição de e-mails legítimos enviados por sistemas não listados no SPF, como plataformas de disparo, sistemas de helpdesk ou integrações com terceiros. O período de monitoramento com ~all é essencial para evitar esse problema.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail, cobrindo SPF, DKIM e DMARC, com explicações em português e orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth
Google — SPF	Guia prático do Google para configuração da política de falha do SPF, com instruções detalhadas e exemplos de configuração	support.google.com/a/answer/33786
RFC 7208	Especificação técnica do protocolo SPF, incluindo a definição formal das políticas de falha hardfail, softfail e neutral e seus efeitos na entrega de e-mails	datatracker.ietf.org/doc/html/rfc7208
MXToolbox — SPF Lookup	Ferramenta para verificação da política de falha configurada no registro SPF do domínio, com análise detalhada dos mecanismos e identificação de problemas	mxtoolbox.com/spf.aspx

> 💡 **Dica:** a política SPF e a política DMARC atuam em camadas complementares. Mesmo com `-all` no SPF, sem uma política **DMARC** ativa (`p=quarantine` ou `p=reject`), alguns servidores destinatários podem ignorar a instrução de rejeição do SPF. A proteção máxima contra spoofing é alcançada com **SPF** em `-all` e **DMARC** em `p=reject` configurados simultaneamente.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A alteração da política é feita editando o **registro SPF** já existente no **DNS**, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- O principal cuidado é garantir que todos os **remetentes legítimos** estejam mapeados antes de aplicar o **hardfail**, para evitar bloqueio de e-mails legítimos.

🛡️ Benefício estimado: Alto

- A política `-all` instrui servidores destinatários a **rejeitar ativamente** e-mails de remetentes não autorizados, complementando a proteção do **DMARC** e fechando a principal porta de entrada para **spoofing do domínio**.

5.6 STARTTLS Disponível

O que é?

O STARTTLS é um comando que permite elevar uma conexão de e-mail inicialmente insegura para uma conexão criptografada com TLS. Quando dois servidores de e-mail se comunicam, o STARTTLS permite que a transmissão das mensagens ocorra de forma criptografada, protegendo o conteúdo contra interceptação durante o trajeto.

> 💡 **Analogia:** sem STARTTLS, e-mails trafegam entre servidores como cartões-postais — visíveis a qualquer intermediário com acesso à rede. Com STARTTLS, o conteúdo é colocado dentro de um envelope lacrado antes do envio. O TLS implícito vai além: exige que o envelope já esteja lacrado antes mesmo de o correio aceitar a encomenda.

> ℹ️ **Nota técnica:** o STARTTLS é considerado uma abordagem de segurança **oportunista**: a criptografia é negociada após o estabelecimento da conexão, o que significa que, em condições adversas ou sob ataque, a conexão pode recair para o modo não criptografado.
>
> É importante distinguir dois trechos do caminho de uma mensagem, porque a recomendação técnica é diferente em cada um:

> • **Entre servidores (entrega para os servidores MX, porta `25`):** não existe padrão de TLS implícito. O transporte continua sendo feito com STARTTLS, e a proteção contra ataques de rebaixamento é obtida com **DANE** (RFC 7672) e **MTA-STS** (RFC 8461), que informam ao servidor remetente que o TLS é obrigatório para aquele domínio.
>
> • **Entre o usuário e o servidor (envio de mensagens e acesso à caixa postal):** a recomendação da **RFC 8314** é usar **TLS implícito**, ou seja, conexões já iniciadas diretamente como TLS, sem negociação prévia — portas `465` para submissão, `993` para IMAP e `995` para POP3.
>
> Este controle avalia o primeiro caso: o suporte a STARTTLS nos servidores MX do domínio.

Como é verificado?

O teste verifica se os servidores de recebimento de e-mail (MX) do domínio oferecem suporte ao STARTTLS.

Riscos da ausência

Risco	Consequência
Interceptação de conteúdo	E-mails trafegam em texto claro entre servidores, podendo ser lidos por qualquer terceiro com acesso à rede
Adulteração em trânsito	Sem criptografia, o conteúdo dos e-mails pode ser modificado durante o trajeto entre servidores
Ausência de autenticação mútua	Sem TLS, não há como verificar a identidade do servidor destinatário, abrindo espaço para servidores impostores

Como resolver?

- Configurar todos os **servidores MX** do domínio para oferecer **STARTTLS**, com **certificado TLS válido** e versões seguras do protocolo (TLS 1.2 e 1.3)
- Não exigir TLS de forma incondicional no recebimento: um servidor MX que rejeita remetentes sem suporte a STARTTLS pode deixar de receber mensagens legítimas de sistemas antigos. O caminho recomendado é oferecer STARTTLS a todos e

sinalizar aos remetentes que o TLS é obrigatório por meio de **DANE** — ver controle 5.7 (Existência de DANE para E-mail) — e de **MTA-STS**

- Nas conexões dos **usuários** ao servidor (envio de mensagens e acesso à caixa postal), adotar **TLS implícito** nas portas `465`, `993` e `995`, conforme a **RFC 8314**

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail e segurança no transporte de mensagens, com orientações práticas para administradores de domínios brasileiros	top.nic.br/faqs/mailauth
RFC 3207	Especificação técnica do STARTTLS para SMTP, definindo o mecanismo de elevação de conexão insegura para TLS	datatracker.ietf.org/doc/html/rfc3207
RFC 8314	Especificação que recomenda o uso de TLS implícito nas conexões dos usuários ao servidor de e-mail (submissão de mensagens, IMAP e POP3), em substituição ao modelo oportunista do STARTTLS	datatracker.ietf.org/doc/html/rfc8314
RFC 8461 — MTA-STS	Especificação do MTA-STS (SMTP MTA Strict Transport Security), mecanismo que permite ao domínio declarar que a entrega de mensagens aos seus servidores MX exige TLS, impedindo ataques de rebaixamento	datatracker.ietf.org/doc/html/rfc8461
MXToolbox — MX Lookup	Ferramenta para verificação dos servidores MX do domínio e análise do suporte a STARTTLS	mxtoolbox.com/mx.aspx
Internet.nl — Mail Test	Ferramenta de análise de segurança de e-mail, incluindo verificação de STARTTLS, versão do TLS e configuração dos servidores MX	internet.nl/mail

💡 **Dica:** o Internet.nl é a ferramenta mais completa para análise de segurança de e-mail, pois verifica simultaneamente STARTTLS, versão do TLS, DANE, SPF, DKIM e DMARC em uma única consulta, oferecendo uma visão consolidada da conformidade do domínio com os padrões modernos de segurança de e-mail.

⚠️ **Atenção:** o STARTTLS por si só não garante que a criptografia será efetivamente utilizada, pois a negociação pode ser suprimida por um atacante posicionado entre os servidores (ataque de downgrade). A proteção contra esse tipo de ataque exige um mecanismo que informe ao servidor remetente que o TLS é obrigatório para aquele domínio. Há dois mecanismos com essa finalidade, complementares entre si:

• **DANE**, avaliado pelo TOP e detalhado nos controles 5.7 (Existência de DANE para E-mail), 5.8 (Validade de DANE para E-mail) e 5.9 (Esquema de Substituição de DANE) deste manual;

• **MTA-STS** (RFC 8461), que publica a mesma exigência por meio de um registro DNS e de um arquivo de política acessível via HTTPS. O MTA-STS não é avaliado pelo TOP e, portanto, não constitui um controle deste manual, mas é recomendado como camada adicional — especialmente em domínios que ainda não possuem DNSSEC, pré-requisito do DANE.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- O suporte a **STARTTLS** é nativo nos principais servidores de e-mail e não requer aquisição de produtos ou licenças adicionais.

🔧 Custo operacional estimado: Baixo

- A habilitação do STARTTLS resume-se à **configuração do servidor de e-mail** e à instalação de um **certificado TLS válido**. O certificado pode ser obtido gratuitamente via **Let's Encrypt**.

🛡️ Benefício estimado: Alto

- A ausência de STARTTLS expõe o **conteúdo de todos os e-mails** trafegados entre servidores, incluindo informações potencialmente sensíveis de comunicações institucionais. A proteção oferecida é **imediata e abrangente**, justificando o benefício alto.

5.7 Existência de DANE (E-mail)

O que é?

O DANE para e-mail funciona com o mesmo princípio descrito no controle 3.15 (Existência de DANE), mas aplicado aos servidores de e-mail (MX): vincula o certificado TLS do servidor de e-mail diretamente ao DNS do domínio, por meio de um **registro** `TLSA`, reduzindo a dependência exclusiva das Autoridades Certificadoras.

> 💡 **Analogia:** o DANE para e-mail é como a carteira de identidade do servidor de correspondência: além do certificado TLS (equivalente ao crachá), há uma validação extra diretamente no DNS do domínio (equivalente ao cadastro oficial) que confirma que o servidor é quem diz ser — mesmo que o crachá tenha sido emitido por uma autoridade comprometida.

> ℹ️ **Nota:** DANE é significativamente mais relevante para e-mail do que para web. Diferentemente da maioria dos navegadores, que ainda não validam DANE para HTTPS, os principais servidores de e-mail modernos, como Postfix e Exchange, já implementam a validação de DANE para SMTP, tornando este controle mais eficaz no contexto de e-mail.

Como é verificado?

O teste verifica se os servidores de nomes de cada servidor MX do domínio possuem um registro `TLSA` válido para DANE.

Riscos da ausência

Risco	Consequência
Certificado falso aceito	Um servidor de e-mail com certificado falso emitido por uma AC comprometida pode interceptar e-mails sem que o servidor remetente perceba
Ataque MitM em SMTP	Um invasor posicionado entre dois servidores de e-mail pode interceptar e modificar mensagens em trânsito
Spoofing de servidor MX	Sem DANE, não há verificação adicional da identidade do servidor MX além do certificado TLS convencional

Como resolver?

1. Garantir que o **DNSSEC** esteja configurado no domínio (pré-requisito)
2. Publicar **registros** `TLSA` para cada **servidor MX** do domínio
3. Manter os **registros atualizados** em sincronia com os **certificados** dos servidores

> ⚠️ **Atenção:** assim como descrito no controle 3.16 (Validade de DANE), ao renovar o certificado de um servidor MX, o registro `TLSA` correspondente deve ser atualizado antes da troca do certificado, seguindo a sequência recomendada de transição com dois registros simultâneos. A desatualização do TLSA pode causar rejeição de e-mails por servidores que validam DANE.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail e segurança no transporte de mensagens, com orientações sobre DANE para servidores MX	top.nic.br/faqs/mailauth

Referência	Descrição	URL
RFC 7672	Especificação técnica do DANE para SMTP, definindo o uso de registros `TLSA` para autenticação de servidores de e-mail	datatracker.ietf.org/doc/html/rfc7672
RFC 6698	Especificação técnica do protocolo DANE e do formato do registro `TLSA`, base para a implementação em servidores MX	datatracker.ietf.org/doc/html/rfc6698
DNSViz	Ferramenta visual de diagnóstico DNSSEC e DANE, permitindo verificar a validade dos registros `TLSA` dos servidores MX e sua integração com a cadeia DNSSEC	dnsviz.net
Internet.nl — Mail Test	Ferramenta de análise de segurança de e-mail que verifica a presença e validade dos registros `TLSA` para os servidores MX do domínio, entre outros controles	internet.nl/mail

💡 **Dica:** o Internet.nl verifica simultaneamente DANE, STARTTLS, SPF, DKIM e DMARC em uma única consulta, sendo a ferramenta mais eficiente para validar a conformidade completa do domínio com os padrões modernos de segurança de e-mail. O DNSViz é complementar ao Internet.nl para diagnóstico visual detalhado da cadeia DNSSEC e dos registros `TLSA` de cada servidor MX.

⚠️ **Atenção para domínios com múltiplos servidores MX:** cada servidor MX deve ter seu próprio registro `TLSA` publicado no DNS. Um domínio com três servidores MX, por exemplo, requer três registros `TLSA` distintos, um para cada servidor. A ausência do registro `TLSA` para qualquer um dos servidores MX pode causar falhas de entrega de e-mail por servidores remetentes que validam DANE estritamente.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo ou Baixo

- Sem custo direto além do já incorrido com **DNSSEC**. Eventual migração de provedor DNS para um compatível com **registros** `TLSA` pode gerar custo marginal.

🔧 Custo operacional estimado: Médio

- A configuração e manutenção dos **registros** `TLSA` para múltiplos **servidores MX** exige conhecimento técnico específico e atenção contínua durante **renovações de certificados**.

🛡️ Benefício estimado: Médio

- Apesar de mais relevante no contexto de e-mail do que no contexto web, o **DANE** ainda não tem adoção completa globalmente, reduzindo seu impacto prático. Seu benefício cresce proporcionalmente à adoção pelo ecossistema de **servidores de e-mail**.

5.8 Validade de DANE (E-mail)

O que é?

Assim como no controle 3.16 (Validade de DANE), a existência do registro `TLSA` não é suficiente: ele precisa ser válido, ou seja, o hash publicado no DNS deve corresponder ao certificado atualmente em uso no servidor de e-mail.

> 💡 **Analogia:** é como um lacre de segurança em documentos oficiais: de nada adianta exigir que toda correspondência venha lacrada se o lacre do destinatário está desatualizado no cadastro oficial. Um **registro** `TLSA` com hash desatualizado é um lacre inválido — existe no papel, mas não oferece proteção real.

Como é verificado?

O teste verifica se as assinaturas DANE dos servidores MX são válidas e correspondem aos certificados TLS atualmente em uso.

Riscos do não atendimento

Risco	Consequência
Proteção nula	Um registro `TLSA` existente, mas com hash desatualizado, não oferece proteção real: servidores que validam DANE ignoram o certificado apresentado por não corresponder ao registrado no DNS
Interrupção do serviço de e-mail	Servidores de e-mail que exigem DANE válido rejeitam mensagens do domínio, causando falhas silenciosas na entrega de e-mails legítimos

Como resolver?

Manter o **registro** `TLSA` sempre **sincronizado** com o **certificado ativo** no **servidor MX**, adotando o esquema de **dois registros simultâneos** durante transições — ver controle 5.9 (Esquema de Substituição de DANE).

1. Gerar o novo certificado (ainda não ativo no servidor MX)
2. Calcular o hash TLSA do novo certificado
3. Publicar o novo registro `TLSA` no DNS
   mantendo o registro `TLSA` do certificado atual
4. Aguardar a propagação DNS (TTL do registro `TLSA`)
5. Substituir o certificado no servidor MX
6. Verificar a validade do DANE com Internet.nl ou DNSViz
7. Remover o registro `TLSA` do certificado antigo

⚠️ **Atenção:** diferentemente de um site fora do ar, falhas na entrega de e-mail são frequentemente silenciosas — o remetente pode não receber nenhuma notificação de erro imediata, e mensagens importantes podem ser perdidas sem que o administrador perceba. A verificação da validade do DANE após qualquer renovação de certificado é essencial para evitar esse cenário.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail e segurança no transporte de mensagens, com orientações sobre manutenção de registros `TLSA` para servidores MX	top.nic.br/faqs/mailauth
RFC 7672	Especificação técnica do DANE para SMTP, incluindo requisitos de validade do registro `TLSA` e procedimentos de atualização	datatracker.ietf.org/doc/html/rfc7672
RFC 7671	Boas práticas para implementação do DANE, com orientações específicas sobre gestão do ciclo de vida do registro `TLSA` durante renovações de certificado	datatracker.ietf.org/doc/html/rfc7671
DNSViz	Ferramenta visual de diagnóstico DANE que permite verificar se o hash no registro `TLSA` corresponde ao certificado atual do servidor MX, identificando divergências	dnsviz.net
Internet.nl — Mail Test	Ferramenta de análise de segurança de e-mail que verifica a validade dos registros `TLSA` para os servidores MX do domínio, identificando divergências entre o hash registrado e o certificado em uso	internet.nl/mail

💡 **Dica:** execute uma verificação no Internet.nl e no DNSViz imediatamente após qualquer renovação de certificado nos servidores MX. Essas ferramentas identificam divergências entre o hash registrado no TLSA e o certificado em uso antes que causem falhas de entrega de e-mail. Para domínios com renovação automática de certificados, é essencial integrar a atualização do registro `TLSA` ao processo de renovação, pois ferramentas como o **Certbot** não atualizam automaticamente os registros `TLSA` no DNS.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A atualização do registro `TLSA` é feita na **zona DNS**, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Médio

- A **sincronização** entre **renovações de certificado** e atualizações de **registros** `TLSA` exige processo operacional bem definido, preferencialmente automatizado, para evitar divergências que causem interrupção do serviço de e-mail.

🛡️ Benefício estimado: Baixo

- O benefício é classificado como baixo não porque a validade do DANE seja irrelevante, mas porque este controle não adiciona proteção nova: ele garante que a proteção já implementada no **controle 5.7** (Existência de DANE para E-mail) permaneça ativa e funcional ao longo do tempo. Em outras palavras, manter o **DANE** válido é a condição mínima para que o investimento já realizado na **implementação do DANE** produza qualquer efeito prático de segurança. Sem essa manutenção, o DANE existe formalmente, mas não protege.

5.9 Esquema de Substituição de DANE

O que é?

Este controle trata especificamente da gestão da transição entre certificados nos servidores de e-mail. Quando um certificado é renovado, há um período de transição durante o qual o registro `TLSA` precisa ser atualizado. Se essa transição não for gerenciada corretamente, o DANE ficará inválido durante o período de transição, causando rejeição de e-mails por servidores que validam DANE. A solução é manter dois registros `TLSA` simultaneamente durante a transição:

> 💡 **Analogia:** é como a renovação de um documento de identidade: o correto é registrar a nova carteira no sistema antes de invalidar a antiga, garantindo continuidade. Se você jogar a antiga fora antes de o sistema reconhecer a nova, há um período em que nenhuma delas é válida — e durante esse período, sua identidade não pode ser confirmada.

> ⚠️ **Atenção:** o registro `TLSA` do novo certificado deve ser publicado com antecedência suficiente para que a propagação do DNS seja completada antes da troca do certificado, geralmente 48 a 72 horas.

Como é verificado?

O teste verifica se há pelo menos dois registros `TLSA` ativos para os servidores MX, indicando que o esquema de substituição está em vigor.

Riscos da ausência

Risco	Consequência
Janela de invalidade	Durante a troca do certificado, o DANE fica inválido, e servidores que validam DANE rejeitam e-mails do domínio
Interrupção do serviço de e-mail	E-mails legítimos enviados por servidores que exigem DANE válido são rejeitados durante o período de transição

Como resolver?

Adotar o processo de **publicação antecipada** do **novo registro** `TLSA` antes de cada **renovação de certificado**, seguindo a sequência:

48 a 72 horas antes da troca

1. Publicar novo registro `TLSA` no DNS
   mantendo o registro `TLSA` do certificado atual
2. Aguardar propagação DNS completa
3. Substituir o certificado no servidor MX
4. Verificar validade do DANE (Internet.nl / DNSViz)
5. Remover o registro `TLSA` do certificado antigo

Transição concluída sem janela de invalidade

Preferencialmente, automatizar esse processo integrando a **atualização do TLSA** ao processo de **renovação do certificado**, garantindo que a sequência correta seja seguida a cada renovação.

⚠️ **Atenção:** ferramentas de renovação automática de certificados, como o Certbot, não atualizam automaticamente os registros `TLSA` no DNS. É necessário integrar explicitamente a atualização do TLSA ao processo de renovação, seja por script personalizado ou por solução de automação DNS que suporte DANE. A ausência dessa integração é a causa mais comum de invalidade de DANE após renovações automáticas de certificado.

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — Autenticação de E-mail	Perguntas frequentes do NIC.br sobre autenticação de e-mail e segurança no transporte de mensagens, com orientações sobre gestão de registros `TLSA` durante renovações	top.nic.br/faqs/mailauth
RFC 7671	Boas práticas para implementação do DANE, com orientações específicas sobre o esquema de dois registros `TLSA` simultâneos durante transições de certificado	datatracker.ietf.org/doc/html/rfc7671
RFC 7672	Especificação técnica do DANE para SMTP, com requisitos sobre manutenção e atualização de registros `TLSA` para servidores MX	datatracker.ietf.org/doc/html/rfc7672
DNSViz	Ferramenta visual de diagnóstico DANE que permite verificar a presença dos dois registros `TLSA` simultâneos e sua correspondência com os certificados em uso	dnsviz.net
Internet.nl — Mail Test	Ferramenta de análise de segurança de e-mail que verifica o esquema de substituição DANE, identificando a presença e validade dos registros `TLSA` durante transições	internet.nl/mail

💡 **Dica:** configure o **TTL** (Time to Live) do registro `TLSA` para um valor baixo — entre **300 e 3600 segundos** — para reduzir o tempo de propagação DNS durante as transições. Um TTL baixo garante que alterações no registro `TLSA` se propaguem mais rapidamente pela internet, diminuindo o período de risco durante a troca de certificados. Lembre-se de que o TTL reduzido aumenta o volume de consultas DNS ao servidor autoritativo, o que deve ser considerado em ambientes com alto volume de consultas.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- A publicação de **registros adicionais no DNS** não gera custo financeiro envolvido.

🔧 Custo operacional estimado: Médio

- O custo operacional é médio porque exige um processo bem definido e disciplinado de gestão do **ciclo de vida dos certificados**, coordenando **renovações** com atualizações de DNS. A **automação** desse processo, embora reduza o esforço contínuo, demanda investimento inicial de configuração.

🛡️ Benefício estimado: Baixo

- O benefício isolado é baixo, pois este controle não adiciona proteção nova, mas preserva a proteção já existente durante as transições de certificado. Sem ele, os controles **5.7** (Existência de DANE para E-mail) e **5.8** (Validade de DANE para E-mail) perdem eficácia periodicamente a cada **renovação de certificado**.

6. Governança do E-mail Institucional

6. Governança do E-mail Institucional

O que é?

Este controle trata de uma questão de governança fundamental: o e-mail oficial de uma organização pública deve estar vinculado a um domínio próprio, sob gestão direta da área de TI da instituição. A utilização de provedores de e-mail genéricos (ex.: '@gmail.com' ou '@hotmail.com') em comunicações oficiais constitui grave vulnerabilidade estrutural. Essa prática compromete diretamente a segurança da informação, a governança corporativa, a continuidade administrativa e a conformidade legal da organização.

💡 **Analogia:** utilizar provedores genéricos (como '@gmail.com') para comunicações oficiais é como um servidor público atender o cidadão entregando um cartão de visita pessoal, sem o timbre da repartição. O endereço identifica o indivíduo, mas não o órgão público. Além disso, em caso de desligamento do servidor, a administração perde totalmente o acesso àquele canal, incluindo o histórico de e-mails e os contatos.

Padrões de utilização recorrentes

Há dois padrões recorrentes de uso de e-mail sem domínio institucional:

Padrão	Exemplo	Problema
Conta pessoal de servidor	joao.silva@gmail.com	Nenhum controle institucional sobre a conta
Conta com aparência institucional	secretaria.educacao@hotmail.com	Aparenta ser oficial, mas está fora do controle da organização

⚠️ **Atenção:** o segundo padrão é especialmente preocupante: embora aparente caráter institucional perante cidadãos e parceiros, a conta pertence ao provedor

> externo e, frequentemente, ao servidor que a criou, não à organização.

Riscos da ausência de domínio institucional próprio

🔐 Segurança da Informação

- Inviabiliza controles técnicos essenciais: sem domínio próprio, é impossível implementar SPF, DKIM e DMARC (controles 5.1 (Existência de DMARC) a 5.5 (Política de SPF)), eliminando toda a camada de proteção contra falsificação de e-mails institucionais
- Impossibilidade de auditoria: a organização não tem acesso a registros detalhados de eventos de segurança, tentativas de acesso suspeitas ou logs das contas utilizadas para comunicações oficiais
- Ausência de políticas institucionais de segurança: controles como bloqueio de anexos maliciosos (CIS v8, Control 9.6) e proteção contra malware no e-mail (CIS v8, Control 9.7) dependem de gestão direta sobre o serviço de e-mail

👤 Governança e Controle de Acesso

- Violação do princípio do privilégio mínimo: a organização não pode revogar acessos de forma tempestiva quando um servidor é desligado, podendo resultar na manutenção indevida de acesso a informações institucionais por pessoas sem vínculo atual
- Ausência de gestão do ciclo de vida das contas: criação, configuração, monitoramento e desativação de contas ficam fora do controle da organização, contrariando o NIST SP 800-53 r5, controles AC-2 (Account Management) e AC-6 (Least Privilege)
- Risco de continuidade administrativa: informações institucionais podem estar retidas em contas pessoais inacessíveis à organização após o desligamento do titular

🏛️ Identidade Institucional e Confiança

- Comprometimento da autenticidade das comunicações: cidadãos, fornecedores e outros órgãos recebem comunicações oficiais originadas em domínios de terceiros, dificultando a identificação da origem legítima e enfraquecendo a confiança nos canais digitais da administração pública

- Superfície de ataque ampliada: comunicações oficiais oriundas de domínios genéricos são mais facilmente imitadas por atacantes em campanhas de phishing, pois qualquer pessoa pode criar uma conta com aparência similar

⚖️ Conformidade Legal

- LGPD (Lei nº 13.709/2018): ao permitir que dados pessoais sejam transmitidos e armazenados em contas fora de sua gestão direta, a organização enfrenta sérias dificuldades para demonstrar o controle e a proteção exigidos pelos arts. 6º, inciso X, e 46 da LGPD, ampliando os riscos de responsabilização em casos de incidentes

> ℹ️ **Importante:** provedores públicos de e-mail possuem mecanismos próprios de segurança, mas voltados ao usuário final genérico, não às necessidades institucionais, ao perfil de risco específico ou às obrigações legais de organizações públicas. A ausência de autonomia para auditar, ajustar e integrar esses controles ao modelo de governança da entidade limita estruturalmente a efetividade das medidas de segurança.

Como resolver?

- Adotar domínio institucional próprio para todas as contas de e-mail utilizadas em comunicações oficiais
- Migrar contas existentes em provedores genéricos para o domínio institucional, com plano de transição que preserve o histórico de comunicações relevantes
- Implementar políticas de gestão de contas cobrindo criação, monitoramento, revisão periódica e desativação imediata em casos de desligamento, conforme NIST SP 800-53 r5, AC-2
- Habilitar os controles técnicos de e-mail descritos na seção 5 deste documento, SPF, DKIM e DMARC, que só são viáveis com domínio institucional próprio
- Estabelecer política institucional que proíba explicitamente o uso de contas em domínios externos para comunicações oficiais

📚 Referências e documentação oficial

Referência	Descrição	URL
LGPD — Lei nº 13.709/2018	Lei Geral de Proteção de Dados Pessoais, com requisitos de controle e proteção de dados pessoais tratados pela organização, incluindo os transmitidos por e-mail	planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm
NIST SP 800-53 r5 — AC-2 e AC-6	Controles de gerenciamento de contas (AC-2) e privilégio mínimo (AC-6) do NIST, com requisitos para criação, monitoramento, revisão, desativação de contas e restrição de acessos ao mínimo necessário para o exercício das funções	csrc.nist.gov/publications/detail/sp/800-53/rev-5/final
CIS Controls v8	Controles de segurança do CIS, incluindo controles 9.6 e 9.7 sobre proteção de e-mail corporativo contra anexos maliciosos e malware	cisecurity.org/controls/v8
Registro.br — Domínios	Informações sobre registro e gestão de domínios institucionais brasileiros, incluindo domínios .gov.br para órgãos públicos	registro.br/ajuda/registro-de-novos-dominios/#registrar-minha-cidade-gov-br

💡 **Dica:** para órgãos públicos federais, estaduais e municipais, o domínio **.gov.br** é o padrão estabelecido para comunicações oficiais e está disponível exclusivamente para entidades governamentais, conferindo imediata identificação institucional e confiabilidade perante cidadãos e parceiros. O Registro.br é o responsável pelo registro e gestão desses domínios no Brasil.

⚠️ **Atenção para processos de migração:** a transição de contas em provedores genéricos para o domínio institucional deve ser planejada com antecedência, incluindo comunicação prévia a todos os contatos relevantes, configuração de redirecionamento temporário das contas antigas e preservação do histórico de comunicações com valor administrativo ou legal. A migração abrupta sem planejamento pode resultar em perda de comunicações relevantes e interrupção de processos em andamento.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo a Médio

- Soluções de e-mail corporativo sob domínio próprio estão disponíveis em uma ampla faixa de custo, desde soluções gratuitas e de código aberto, como **Zimbra** e **Postfix**, até serviços gerenciados em nuvem, como **Google Workspace for Government** e **Microsoft 365**, com custo por usuário por mês. O custo financeiro varia significativamente conforme o número de usuários e a solução escolhida, mas é plenamente justificado pelos riscos eliminados. Organizações que já utilizam plataformas de nuvem governamentais podem ter esse recurso disponível sem custo financeiro envolvido.

🔧 Custo operacional estimado: Médio

- O custo operacional é médio porque: a **migração de contas** existentes em provedores genéricos para o domínio institucional exige planejamento, comunicação interna e um período de transição gerenciado.
- A gestão contínua do serviço de e-mail institucional, incluindo criação e desativação de contas, políticas de segurança e monitoramento, demanda capacidade operacional da área de TI que pode não estar dimensionada para isso em organizações menores.
- A implementação dos controles técnicos de e-mail, **SPF**, **DKIM** e **DMARC**, após a adoção do domínio próprio, adiciona esforço inicial de configuração, já detalhado na seção 5.

🛡️ Benefício estimado: Alto

- O benefício é alto porque: este é o único controle desta seção que é **pré-requisito** para todos os demais controles de e-mail: sem domínio institucional próprio, **SPF**, **DKIM**, **DMARC** e **DANE** são inviáveis, tornando toda a proteção da seção 5 inaplicável.
- A adoção do domínio próprio elimina simultaneamente riscos de **segurança**, de **governança**, de **continuidade administrativa** e de **conformidade legal**, sendo o controle de maior impacto transversal desta seção.
- Protege diretamente cidadãos, que passam a receber comunicações oficiais de endereços verificáveis e autênticos, e a organização, que passa a ter **controle efetivo** sobre suas comunicações institucionais.

7. Modernização de Endereçamento (IPv6)

O IPv6 é o sucessor do IPv4, desenvolvido para superar a limitação de endereços disponíveis no protocolo anterior. Enquanto o IPv4 suporta cerca de 4 bilhões de endereços, o IPv6 suporta um número praticamente ilimitado, além de trazer melhorias em desempenho, segurança e simplicidade de configuração.

⚠️ O esgotamento de endereços IPv4 é uma realidade consolidada. Provedores de internet e usuários que operam exclusivamente em IPv6 já representam uma parcela significativa e crescente do tráfego global. Serviços que não oferecem suporte a IPv6 tornam-se progressivamente inacessíveis para esse público.

Os controles desta seção verificam se toda a cadeia de serviços do domínio, incluindo servidores de nomes DNS, servidor web e servidores de e-mail, está corretamente configurada e acessível via IPv6, em paridade com o IPv4.

7.1 Endereços IPv6 para Servidores de Nomes

O que é?

Os servidores de nomes (NS) são responsáveis por responder às consultas DNS do domínio. Para que sejam acessíveis a clientes que operam exclusivamente em IPv6, cada servidor de nomes deve ter um endereço IPv6 associado, publicado como registro `AAAA` no DNS.

💡 **Analogia:** é como um serviço de informações que só atende ligações pelo telefone fixo: quem liga pelo aparelho novo (IPv6) simplesmente não consegue chegar ao atendimento. E sem conseguir resolver o nome do domínio, todos os serviços do endereço — site, e-mail, APIs — ficam inacessíveis para esse público.

Como é verificado?

O teste verifica se o domínio possui pelo menos dois servidores de nomes com endereço IPv6 configurado.

Riscos da ausência

Risco	Consequência
Inacessibilidade para usuários IPv6	Clientes que operam exclusivamente em IPv6 não conseguem resolver o domínio, tornando todos os serviços do domínio inacessíveis para esse público
Desempenho reduzido	Clientes com IPv6 preferencial podem enfrentar latência adicional ao recorrer ao IPv4 como alternativa
Incompatibilidade futura	À medida que a adoção de IPv6 cresce, a ausência de suporte torna-se uma barreira progressivamente maior de acessibilidade

Como resolver?

1. Verificar se o **provedor DNS** atual suporta **endereços IPv6** para servidores de nomes
2. Publicar **registros** `AAAA` para cada servidor de nomes do domínio
3. Garantir que **pelo menos dois servidores de nomes** tenham endereços IPv6, mantendo a **redundância**

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — IPv6	Portal do NIC.br dedicado ao IPv6 no Brasil, com documentação técnica, guias de implementação e estatísticas de adoção	ipv6.br
TOP — Teste de Conexão	Módulo da plataforma TOP que verifica o suporte a IPv6 de domínios brasileiros, incluindo servidores de nomes, servidor web e servidores de e-mail	top.nic.br/test-connection
RFC 3596	Especificação técnica do registro `AAAA` no DNS, utilizado para associar endereços IPv6 a nomes de domínio, incluindo servidores de nomes	datatracker.ietf.org/doc/html/rfc3596

Referência	Descrição	URL
Internet.nl	Ferramenta de análise de conformidade com padrões modernos de internet, incluindo verificação de IPv6 para servidores de nomes, web e e-mail	internet.nl
MXToolbox — DNS Lookup	Ferramenta para verificação dos registros DNS do domínio, incluindo registros `AAAA` dos servidores de nomes	mxtoolbox.com/SuperTool.aspx

> 💡 **Dica:** o TOP — Teste de Conexão (top.nic.br/test-connection) verifica simultaneamente o suporte a IPv6 de toda a cadeia de serviços do domínio — servidores de nomes, servidor web e servidores de e-mail — em uma única consulta, sendo a ferramenta mais prática para verificar os controles desta seção de forma consolidada.

> ⚠️ **Atenção:** a configuração de IPv6 nos servidores de nomes depende do suporte oferecido pelo provedor DNS contratado. Provedores DNS que não oferecem endereços IPv6 para seus servidores de nomes inviabilizam o atendimento deste controle independentemente da configuração do domínio. Caso o provedor atual não suporte IPv6, avaliar a migração para um provedor que ofereça suporte completo a IPv6 é a solução mais adequada.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo

- O custo financeiro é baixo porque: equipamentos de rede adquiridos nos últimos 10 a 15 anos, como **roteadores**, **firewalls** e **switches**, já oferecem suporte a **IPv6** nativamente, sem necessidade de substituição de hardware.
- Os principais **provedores DNS**, sejam comerciais ou de nuvem, oferecem suporte a IPv6 como recurso padrão, tornando improvável a necessidade de migração de provedor por esse motivo.

- Em provedores de nuvem, o suporte a **IPv6** está disponível sem custo financeiro envolvido na maioria dos casos.

🔧 Custo operacional estimado: Alto

- A implementação de **IPv6** em toda a infraestrutura requer planejamento e execução cuidadosos, incluindo reconfiguração de **redes**, **firewalls**, **regras de segurança** e monitoramento. A equipe técnica pode necessitar de **capacitação específica em IPv6**, dado que o protocolo apresenta diferenças operacionais relevantes em relação ao IPv4. O processo de transição deve ser gerenciado para garantir que os serviços permaneçam disponíveis em **IPv4** durante e após a migração.

🛡️ Benefício estimado: Médio

- O benefício é médio porque: a inacessibilidade por **IPv6** afeta diretamente apenas usuários e sistemas que operam exclusivamente nesse protocolo, ainda uma minoria, embora crescente.
- No entanto, a tendência de **adoção de IPv6** é irreversível, e o custo de implementação cresce com o adiamento, tornando a **adoção antecipada** estrategicamente vantajosa.
- O benefício imediato é limitado, mas o benefício de longo prazo, em termos de **acessibilidade** e **conformidade com padrões modernos de internet**, é significativo.

7.2 Acessibilidade IPv6 dos Servidores de Nomes

O que é?

Ter um endereço IPv6 publicado no registro `AAAA` não garante que o servidor de nomes esteja efetivamente respondendo a consultas via IPv6. Este controle verifica se os servidores de nomes com endereço IPv6 publicado estão de fato acessíveis e operantes nesse protocolo.

> 💡 **Analogia:** publicar um endereço IPv6 sem o servidor estar ouvindo nesse endereço é como colocar um número de telefone no cartão de visita de uma linha

que nunca é atendida.

Como é verificado?

O teste verifica se todos os servidores de nomes que possuem registro `AAAA` respondem corretamente a consultas DNS via IPv6.

Riscos da ausência

Risco	Consequência
Servidor de nomes fantasma	O endereço IPv6 está publicado no DNS, mas o servidor não responde, causando falhas de resolução DNS para clientes IPv6
Degradação de desempenho	Clientes que tentam o endereço IPv6 e não obtêm resposta sofrem timeout antes de recorrer ao IPv4, aumentando a latência de acesso

Como resolver?

- Verificar se o **servidor de nomes** está configurado para escutar e responder em seu **endereço IPv6**
- Confirmar que as **regras de firewall** permitem tráfego **DNS** (**porta** `53` **UDP e TCP**) via IPv6
- Testar a acessibilidade regularmente com ferramentas de verificação de DNS

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — IPv6	Portal do NIC.br dedicado ao IPv6 no Brasil, com documentação técnica e guias de implementação para servidores de nomes	ipv6.br
TOP — Teste de Conexão	Módulo da plataforma TOP que verifica o suporte e a acessibilidade IPv6 de domínios brasileiros, incluindo testes de resposta dos servidores de nomes	top.nic.br/test-connection

Referência	Descrição	URL
Internet.nl	Ferramenta de análise de conformidade com padrões modernos de internet, incluindo verificação de acessibilidade IPv6 dos servidores de nomes	internet.nl
MXToolbox — DNS Lookup	Ferramenta para verificação de registros DNS e teste de resposta dos servidores de nomes via IPv6	mxtoolbox.com/SuperTool.aspx
DNSViz	Ferramenta visual de diagnóstico DNS que permite verificar a acessibilidade e o funcionamento dos servidores de nomes via IPv6	dnsviz.net

💡 **Dica:** o TOP — Teste de Conexão (top.nic.br/test-connection) e o Internet.nl testam ativamente a acessibilidade dos servidores de nomes via IPv6, verificando não apenas a existência do registro `AAAA`, mas também se o servidor responde efetivamente a consultas DNS nesse protocolo. A diferença entre os controles 7.1 (Endereços IPv6 para Servidores de Nomes) e 7.2 (Acessibilidade IPv6 dos Servidores de Nomes) é precisamente essa: o 7.1 verifica a existência do registro `AAAA`, enquanto o 7.2 verifica se o servidor de fato responde no endereço publicado.

⚠️ **Atenção:** em ambientes com firewall, é comum que o endereço IPv6 esteja corretamente configurado no servidor de nomes, mas que regras de firewall bloqueiem o tráfego DNS na porta `53` via IPv6. Verificar explicitamente as regras de firewall para tráfego IPv6 é tão importante quanto verificar a configuração do servidor de nomes em si. O bloqueio pode ser seletivo — por exemplo, permitindo UDP mas bloqueando TCP na porta `53` — o que causa falhas intermitentes e de difícil diagnóstico.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Trata-se de uma **verificação e ajuste de configuração** no **servidor de nomes** já existente, sem custo financeiro adicional além do já incorrido na implementação do IPv6 — ver controle 7.1 (Endereços IPv6 para Servidores de Nomes).

🔧 Custo operacional estimado: Baixo

- A verificação e correção da **acessibilidade IPv6** resume-se a **ajustes de configuração** e **regras de firewall**, com ferramentas gratuitas de diagnóstico disponíveis.

🛡️ Benefício estimado: Médio

- Um **endereço IPv6 publicado, mas inacessível**, é potencialmente pior do que não ter IPv6: causa **falhas ativas** e **timeout** para clientes IPv6, em vez de simplesmente não atendê-los. Corrigir essa inconsistência é fundamental para garantir que a implementação do IPv6 produza o efeito desejado.

7.3 Endereços IPv6 para Servidor Web

O que é?

A exemplo dos servidores de nomes, o servidor web deve possuir um endereço IPv6 (registro `AAAA`) publicado no DNS. Essa configuração garante o acesso aos usuários conectados exclusivamente por essa versão do protocolo. Sem esse registro, o site torna-se inacessível para esse público.

> 💡 **Analogia:** é como ter uma sede física sem placa na rua IPv6: quem vem pelo caminho novo não encontra a entrada. Para esse público, o serviço simplesmente não existe — não por falha técnica do visitante, mas por ausência de sinalização no endereço correto.

Como é verificado?

O teste verifica se existe pelo menos um registro `AAAA` com endereço IPv6 apontando para o servidor web do domínio.

Riscos da ausência

Risco	Consequência
Inacessibilidade do site	Usuários e sistemas que operam exclusivamente em IPv6 não conseguem acessar o site
Exclusão digital progressiva	Com o crescimento da adoção de IPv6, a ausência de suporte torna o site inacessível para uma parcela crescente do público

Como resolver?

- Obter um **endereço IPv6** para o servidor web junto ao **provedor de hospedagem** ou provedor de internet
- Publicar o **registro** `AAAA` correspondente na **zona DNS** do domínio

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — IPv6	Portal do NIC.br dedicado ao IPv6 no Brasil, com documentação técnica e guias de implementação para servidores web	ipv6.br
TOP — Teste de Conexão	Módulo da plataforma TOP que verifica o suporte a IPv6 de domínios brasileiros, incluindo a existência do registro `AAAA` do servidor web	top.nic.br/test-connection
Internet.nl	Ferramenta de análise de conformidade com padrões modernos de internet, incluindo verificação de IPv6 para servidores web	internet.nl

Referência	Descrição	URL
MXToolbox	Ferramenta para verificação de registros DNS, incluindo a existência e o conteúdo do registro `AAAA` do servidor web	mxtoolbox.com/SuperTool.aspx
RFC 3596	Especificação técnica do registro `AAAA` no DNS, utilizado para associar endereços IPv6 a nomes de domínio	datatracker.ietf.org/doc/html/rfc3596

💡 **Dica:** o TOP — Teste de Conexão (top.nic.br/test-connection) e o Internet.nl verificam simultaneamente o suporte a IPv6 de toda a cadeia de serviços do domínio — servidores de nomes, servidor web e servidores de e-mail — permitindo identificar em uma única consulta quais componentes ainda carecem de configuração IPv6.

⚠️ **Atenção:** a disponibilidade de endereços IPv6 para o servidor web depende do suporte oferecido pelo provedor de hospedagem ou de infraestrutura contratado. Provedores que não oferecem endereços IPv6 inviabilizam o atendimento deste controle independentemente da configuração do domínio. Ao contratar ou renovar serviços de hospedagem, verificar explicitamente o suporte a IPv6 como requisito técnico obrigatório.

Custos e Benefícios

💰 Custo financeiro estimado: Baixo

- A maioria dos **provedores de hospedagem** e nuvem modernos já incluem **endereçamento IPv6** sem custo financeiro envolvido. Em **infraestruturas legadas**, pode haver custo de adequação, mas geralmente inferior ao da implementação nos servidores de nomes.

🔧 Custo operacional estimado: Baixo

- A publicação do **registro** `AAAA` é simples e segue o mesmo processo de um registro `A` convencional. O principal esforço está em garantir que o **servidor web** esteja

configurado para escutar no **endereço IPv6** atribuído.

🛡️ Benefício estimado: Médio

- Pelas mesmas razões do controle 7.1 (Endereços IPv6 para Servidores de Nomes): o impacto imediato é limitado ao público que opera exclusivamente em **IPv6**, mas cresce progressivamente com a **adoção do protocolo**.

7.4 Acessibilidade IPv6 do Servidor Web

O que é?

Assim como no controle 7.2 (Acessibilidade IPv6 dos Servidores de Nomes), ter um registro `AAAA` publicado não garante que o servidor web esteja efetivamente respondendo a requisições HTTP e HTTPS via IPv6. Este controle verifica se o servidor web está operante e acessível nas portas padrão (`80` para HTTP e `443` para HTTPS) via IPv6.

> 💡 **Analogia:** é como um endereço registrado no mapa, mas com a porta trancada: quem chega via IPv6 bate e não recebe resposta, mesmo com o endereço correto publicado no DNS.

Como é verificado?

O teste verifica se é possível estabelecer conexão com o servidor web via IPv6 nas portas `80` ou `443`.

Riscos da ausência

Risco	Consequência
Servidor web inacessível via IPv6	Mesmo com endereço IPv6 publicado, o site não responde a requisições nesse protocolo

Risco	Consequência
Timeout para usuários IPv6	Usuários com IPv6 preferencial enfrentam tentativas frustradas antes de recorrer ao IPv4, aumentando a latência percebida

Como resolver?

- Configurar o **servidor web** para escutar nas **portas** `80` **e** `443` no **endereço IPv6** atribuído
- Verificar se as **regras de firewall** permitem tráfego **HTTP e HTTPS** via IPv6
- Garantir que o **certificado TLS** inclua o endereço IPv6 no campo **SAN**, se aplicável

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — IPv6	Portal do NIC.br dedicado ao IPv6 no Brasil, com documentação técnica e guias de configuração de servidores web para IPv6	ipv6.br
TOP — Teste de Conexão	Módulo da plataforma TOP que verifica a acessibilidade IPv6 do servidor web, testando ativamente a resposta nas portas HTTP e HTTPS	top.nic.br/test-connection
Internet.nl	Ferramenta de análise de conformidade com padrões modernos de internet, incluindo verificação de acessibilidade IPv6 do servidor web nas portas `80` e `443`	internet.nl
MXToolbox	Ferramenta para verificação de conectividade com o servidor web via IPv6, incluindo teste de resposta nas portas padrão	mxtoolbox.com/SuperTool.aspx

> 💡 **Dica:** a diferença entre os controles 7.3 (Endereços IPv6 para Servidor Web) e 7.4 (Acessibilidade IPv6 do Servidor Web) é a mesma entre os controles 7.1 (Endereços IPv6 para Servidores de Nomes) e 7.2 (Acessibilidade IPv6 dos Servidores

de Nomes): o 7.3 verifica a existência do registro `AAAA` no DNS, enquanto o 7.4 verifica se o servidor web de fato responde no endereço publicado. Um servidor pode ter o registro `AAAA` corretamente publicado e ainda assim ser inacessível via IPv6 por falha de configuração no servidor web ou bloqueio de firewall.

⚠️ **Atenção:** em ambientes com balanceadores de carga ou proxies reversos, a configuração de IPv6 deve ser verificada em cada camada da infraestrutura. É comum que o balanceador de carga tenha endereço IPv6 configurado, mas que a comunicação entre o balanceador e os servidores de aplicação ocorra exclusivamente via IPv4 — o que pode causar falhas em cenários específicos. Verificar toda a cadeia de comunicação, do cliente ao servidor de aplicação, é essencial para garantir o suporte completo a IPv6.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo

- Ajuste de configuração no **servidor web** e nas **regras de firewall** já existentes, sem custo financeiro envolvido.

🔧 Custo operacional estimado: Baixo

- A configuração resume-se a garantir que o **servidor web** e o **firewall** estejam corretamente configurados para **IPv6**, processo análogo ao já realizado para **IPv4**.

🛡️ Benefício estimado: Médio

- Um endereço IPv6 publicado sem o servidor respondendo gera **falhas ativas** para usuários IPv6, sendo potencialmente mais prejudicial do que simplesmente não ter IPv6. Garantir a **acessibilidade efetiva** é o que transforma o endereçamento **IPv6** em benefício real.

7.5 Mesmo Site com Endereços IPv6 e IPv4

O que é?

Este controle verifica se o conteúdo e as funcionalidades do site são idênticos quando acessados via IPv4 e via IPv6. Em alguns casos, a implementação de IPv6 é feita de forma incompleta, resultando em um site com menos funcionalidades, conteúdo desatualizado ou configurações de segurança inferiores quando acessado pelo protocolo mais novo.

> 💡 **Analogia:** é como ter duas entradas para o mesmo edifício, mas a entrada IPv6 leva a um andar com menos salas abertas e sem câmeras de segurança. Quem entra por ali tem uma experiência diferente e menos segura.

Como é verificado?

O teste compara o conteúdo retornado pelo servidor web quando acessado via IPv4 e via IPv6, verificando se há diferenças relevantes de conteúdo ou funcionalidade.

Riscos da ausência

Risco	Consequência
Experiência degradada para usuários IPv6	Funcionalidades, serviços ou conteúdos disponíveis em IPv4 ficam inacessíveis para usuários IPv6
Brechas de segurança no acesso IPv6	Um site IPv6 com configurações de segurança inferiores, como ausência de HTTPS ou cabeçalhos de segurança, pode ser explorado por invasores que acessam especificamente por esse protocolo
Inconsistência de informações	Conteúdo desatualizado ou diferente no acesso IPv6 compromete a confiabilidade do serviço

Como resolver?

- Garantir que a **infraestrutura IPv6** espelhe integralmente a **configuração IPv4**, incluindo **configurações TLS**, **cabeçalhos de segurança** e funcionalidades da

aplicação

- Realizar testes periódicos comparando o acesso via IPv4 e IPv6
- Utilizar o mesmo servidor ou **balanceador de carga** para ambos os protocolos, evitando configurações divergentes

📚 Referências e documentação oficial

Referência	Descrição	URL
NIC.br — IPv6	Portal do NIC.br dedicado ao IPv6 no Brasil, com documentação técnica sobre implementação de IPv6 em paridade com IPv4	ipv6.br
TOP — Teste de Conexão	Módulo da plataforma TOP que verifica e compara o comportamento do servidor web quando acessado via IPv4 e via IPv6, identificando diferenças de conteúdo e configuração	top.nic.br/test-connection
Internet.nl	Ferramenta de análise de conformidade com padrões modernos de internet, incluindo verificação de paridade de conteúdo entre acesso IPv4 e IPv6	internet.nl
RFC 4213	Especificação técnica de mecanismos de transição para IPv6, com orientações para manutenção de paridade de serviço entre os dois protocolos	datatracker.ietf.org/doc/html/rfc4213

💡 **Dica:** a forma mais eficaz de garantir paridade entre IPv4 e IPv6 é utilizar a mesma instância de servidor ou balanceador de carga para atender requisições de ambos os protocolos, configurando-o para escutar em ambos os endereços. Essa abordagem elimina por design o risco de configurações divergentes, pois toda alteração feita no servidor se aplica automaticamente a ambos os protocolos.

⚠️ **Atenção:** ambientes que utilizam infraestruturas separadas para IPv4 e IPv6 — como servidores distintos ou configurações de proxy reverso independentes — estão sujeitos a divergências graduais de configuração ao longo do tempo, especialmente após atualizações ou mudanças que sejam aplicadas em apenas uma das infraestruturas. Nesses ambientes, recomenda-se incluir explicitamente a verificação de paridade IPv4/IPv6 nos processos de gestão de mudanças e nas rotinas de monitoramento.

Custos e Benefícios

💰 Custo financeiro estimado: Nulo a Baixo

- Em infraestruturas baseadas em nuvem, a paridade entre **IPv4 e IPv6** geralmente é alcançada com configurações simples e sem custo financeiro envolvido. Em **infraestruturas legadas** com arquiteturas mais complexas, pode ser necessário adequar sistemas e **APIs** que não foram projetados para **IPv6**, gerando custo de desenvolvimento.

🔧 Custo operacional estimado: Médio

- Manter a **paridade de conteúdo e segurança** entre **IPv4 e IPv6** exige que todos os processos de atualização, manutenção e verificação de segurança sejam aplicados a ambos os protocolos de forma consistente, aumentando o escopo das atividades operacionais regulares.

🛡️ Benefício estimado: Médio

- Uma **implementação IPv6 incompleta** pode criar uma **superfície de ataque** adicional, com configurações de segurança inferiores no acesso por esse protocolo. Garantir a **paridade** elimina esse risco e assegura que todos os usuários, independentemente do protocolo utilizado, tenham a mesma experiência segura e completa.

Quadro de Custo e Benefício dos Controles

O quadro a seguir consolida, para cada controle deste manual, a estimativa de custo financeiro, custo operacional e benefício de segurança esperado com a sua implementação. Os valores são estimativas qualitativas destinadas a orientar a priorização das ações, agrupadas pelas seções deste manual.

Escala de desempenho: Nulo (custo) · Alto (benef.) | Baixo (custo) · Médio-Alto (benef.) | Baixo a Médio (custo) · Médio (benef.) | Médio (custo) | Alto (custo) · Baixo (benef.)

Controle	Custo Financeiro	Custo Operacional	Benefício
1 — DNS			
1.1 Existência de DNSSEC	Baixo a Médio	Médio	Alto
1.2 Validade de DNSSEC	Nulo	Baixo	Alto
2 — HTTPS E TRANSPORTE			
2.1 HTTPS Disponível (HTTP/TLS)	Baixo a Médio	Médio	Alto
2.2 Redirecionamento para HTTPS	Nulo	Baixo	Alto
2.3 Ausência de Compressão HTTP	Nulo	Baixo	Médio
2.4 HSTS (HTTP Strict Transport Security)	Nulo	Médio	Alto
3 — TLS E CERTIFICADOS			
3.1 Versão do TLS	Nulo	Médio	Médio
3.2 Cifras (Seleção de Algoritmos)	Nulo	Baixo a Médio	Médio
3.3 Ordem das Cifras	Nulo	Baixo	Médio
3.4 Parâmetros de Troca de Chaves	Nulo	Baixo	Médio
3.5 Função Hash para Troca de Chaves	Nulo ou Baixo	Baixo	Médio
3.6 Compressão TLS	Nulo	Baixo	Médio
3.7 Renegociação Segura	Nulo	Baixo	Médio

Controle	Custo Financeiro	Custo Operacional	Benefício
3.8 Renegociação Iniciada pelo Cliente	Nulo	Baixo	Médio
3.9 0-RTT (Zero Round Trip Time)	Nulo	Baixo a Médio	Médio a Alto
3.10 OCSP Stapling	Nulo	Baixo	Médio a Alto
3.11 Cadeia de Confiança do Certificado	Baixo	Baixo	Alto
3.12 Chave Pública do Certificado	Nulo ou Baixo	Baixo	Médio ou Alto
3.13 Assinatura do Certificado	Nulo ou Baixo	Baixo	Médio ou Alto
3.14 Nome de Domínio no Certificado	Nulo ou Baixo	Baixo	Médio a Alto
3.15 Existência de DANE	Nulo ou Baixo	Médio	Baixo
3.16 Validade de DANE	Nulo	Médio	Baixo
4 — CABEÇALHOS DE SEGURANÇA HTTP			
4.1 `X-Frame-Options`	Nulo	Baixo	Alto
4.2 `X-Content-Type-Options`	Nulo	Baixo	Alto
4.3 `Content-Security-Policy` (CSP)	Nulo	Baixo a Médio	Alto
4.4 `Referrer-Policy`	Nulo	Baixo	Alto
5 — AUTENTICAÇÃO DE E-MAIL			
5.1 Existência de DMARC	Nulo	Baixo	Alto
5.2 Política de DMARC	Nulo	Baixo	Alto
5.3 Existência de DKIM	Nulo	Baixo	Alto
5.4 Existência de SPF	Nulo	Baixo	Alto
5.5 Política de SPF	Nulo	Baixo	Alto
5.6 STARTTLS Disponível	Nulo	Baixo	Alto

Controle	Custo Financeiro	Custo Operacional	Benefício
5.7 Existência de DANE (E-mail)	Nulo ou Baixo	Médio	Médio
5.8 Validade de DANE (E-mail)	Nulo	Médio	Baixo
5.9 Esquema de Substituição de DANE	Nulo	Médio	Baixo
6 — GOVERNANÇA DO E-MAIL INSTITUCIONAL			
6. Governança do E-mail Institucional	Baixo a Médio	Médio	Alto
7 — IPV6			
7.1 Endereços IPv6 para Servidores de Nomes	Baixo	Alto	Médio
7.2 Acessibilidade IPv6 dos Servidores de Nomes	Nulo	Baixo	Médio
7.3 Endereços IPv6 para Servidor Web	Baixo	Baixo	Médio
7.4 Acessibilidade IPv6 do Servidor Web	Nulo	Baixo	Médio
7.5 Mesmo Site com Endereços IPv6 e IPv4	Nulo a Baixo	Médio	Médio

Glossário de Termos Técnicos

Este glossário reúne os principais termos técnicos utilizados neste manual, organizados em ordem alfabética. Cada verbete apresenta uma explicação sucinta orientada ao contexto desta publicação, seguida de referências para aprofundamento.

A

A Record

Registro A

Tipo de registro DNS que associa um nome de domínio a um endereço IPv4. É o tipo de registro DNS mais básico e amplamente utilizado.

→ RFC 1035 — Domain Names — Implementation and Specification: datatracker.ietf.org/doc/html/rfc1035

AAAA Record

Registro AAAA

Tipo de registro DNS que associa um nome de domínio a um endereço IPv6. É o equivalente IPv6 do registro A, que associa nomes a endereços IPv4. A nomenclatura "AAAA" reflete que um endereço IPv6 tem quatro vezes o tamanho de um endereço IPv4 (128 bits contra 32 bits).

→ RFC 3596 — DNS Extensions to Support IP Version 6: datatracker.ietf.org/doc/html/rfc3596

AC — Autoridade Certificadora

Certificate Authority — CA

Entidade responsável por emitir, gerenciar e revogar certificados digitais, atestando a identidade de sites, organizações e indivíduos na internet. As ACs formam uma hierarquia de confiança reconhecida pelos navegadores e sistemas operacionais. Exemplos: ICP-Brasil, Let's Encrypt, DigiCert, Sectigo.

→ RFC 5280 — Internet X.509 PKI Certificate and CRL Profile: datatracker.ietf.org/doc/html/rfc5280

→ NIST SP 800-57 Part 1 Rev. 5: csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-5/final

Algoritmo Criptográfico

Procedimento matemático usado para cifrar, decifrar, assinar digitalmente ou verificar dados. Exemplos relevantes neste manual: RSA, ECDSA e EdDSA (para assinatura), ECDHE e DHE (para troca de chaves), AES e ChaCha20 (para cifragem), SHA-256 e SHA-384 (para funções hash).

→ NIST, Glossary of Key Information Security Terms: csrc.nist.gov/glossary

→ NIST FIPS 140-3, Security Requirements for Cryptographic Modules: csrc.nist.gov/publications/detail/fips/140/3/final

B

BEAST

Browser Exploit Against SSL/TLS

Ataque criptográfico publicado em 2011 que explora uma vulnerabilidade no modo de cifração CBC (Cipher Block Chaining) do TLS 1.0, permitindo que um invasor posicionado entre o cliente e o servidor decifre partes do tráfego criptografado. Mitigado pela adoção do TLS 1.2 ou superior.

→ OWASP, Transport Layer Security Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

→ CVE-2011-3389: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3389

Bleichenbacher / ROBOT

Return Of Bleichenbacher's Oracle Threat

Classe de ataques de oráculo de preenchimento (padding oracle) contra o RSA, descrita originalmente por Daniel Bleichenbacher em 1998. O invasor envia mensagens deliberadamente malformadas ao servidor e, observando as diferenças nas respostas de erro, consegue recuperar a chave de sessão e decifrar a comunicação. Em 2017, a pesquisa ROBOT demonstrou que diversas implementações de TLS amplamente utilizadas continuavam vulneráveis. Mitigado pela adoção do TLS 1.3 e pela remoção das cifras de troca de chaves baseadas em RSA.

→ robotattack.org: robotattack.org

→ CVE-2017-13099: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-13099

BREACH

Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext

Ataque publicado em 2013 que explora a compressão HTTP em conjunto com a criptografia TLS para inferir o conteúdo de dados confidenciais, como cookies de sessão, analisando variações no tamanho das respostas comprimidas. Mitigado pela desativação da compressão HTTP para respostas que contenham dados sensíveis.

→ Site oficial do ataque BREACH: breachattack.com

→ OWASP, Transport Layer Security Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

C

Cadeia de Confiança

Chain of Trust

Hierarquia de certificados digitais que liga o certificado de um site a uma Autoridade Certificadora raiz reconhecida e confiável pelos navegadores e sistemas operacionais. Para que um certificado seja aceito, é necessário que todos os elos da cadeia — certificado do site, certificados intermediários e certificado raiz — sejam válidos e reconhecidos.

→ RFC 5280 — Internet X.509 PKI Certificate and CRL Profile: datatracker.ietf.org/doc/html/rfc5280

→ MDN Web Docs, Certificate Chains: developer.mozilla.org/en-US/docs/Web/Security/Certificate_Transparency

CBC

Cipher Block Chaining

Modo de operação de cifras de bloco em que cada bloco de texto cifrado é combinado com o bloco anterior antes da cifragem, criando dependência entre os blocos. Embora amplamente utilizado, o CBC é vulnerável a ataques como BEAST e POODLE quando combinado com versões antigas do TLS.

→ NIST SP 800-38A, Recommendation for Block Cipher Modes of Operation: csrc.nist.gov/publications/detail/sp/800-38a/final

Certificado Digital

Digital Certificate

Documento eletrônico que associa uma chave pública a uma identidade (pessoa, organização ou domínio), atestado por uma Autoridade Certificadora. Utilizado em HTTPS para garantir a autenticidade do servidor e viabilizar a comunicação criptografada. Os principais tipos utilizados em sites são:

→ **DV (Domain Validation):** valida apenas o controle sobre o domínio.

→ **OV (Organization Validation):** valida também a identidade da organização.

→ **EV (Extended Validation):** valida com rigor adicional a identidade e a existência legal da organização.

→ RFC 5280 — Internet X.509 PKI Certificate and CRL Profile: datatracker.ietf.org/doc/html/rfc5280

→ Federal PKI — Governança e Infraestrutura (substitui NIST SP 800-32): idmanagement.gov/fpki/

Certbot

Ferramenta de linha de comando gratuita e de código aberto, mantida pela Electronic Frontier Foundation (EFF), que automatiza a obtenção, instalação e renovação de certificados digitais emitidos pela Let's Encrypt, com suporte aos principais servidores web.

→ Site oficial: certbot.eff.org

CIS Controls

Center for Internet Security Controls

Conjunto de boas práticas de segurança cibernética publicado pelo Center for Internet Security (CIS), organizado em 18 controles prioritários. O CIS Controls v8 é amplamente adotado como referência para implementação de segurança em organizações de todos os portes.

→ CIS Controls v8: cisecurity.org/controls/v8

Cifra

Cipher Suite

Conjunto de algoritmos criptográficos negociados durante o handshake TLS que define como a comunicação será protegida. Uma cifra completa especifica: o algoritmo de troca de chaves, o algoritmo de autenticação, o algoritmo de cifragem e a função hash para integridade. Exemplo: `TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384` — Troca de chaves: ECDHE; Autenticação: RSA; Cifragem: AES-256-GCM; Hash: SHA-384.

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

→ Mozilla SSL Configuration Generator: ssl-config.mozilla.org

Clickjacking

Ataque em que um invasor sobrepõe uma página web legítima de forma invisível sobre outra página maliciosa, induzindo o usuário a interagir com elementos da página legítima sem perceber. Mitigado pelos cabeçalhos `X-Frame-Options` e `Content-Security-Policy` (diretiva frame-ancestors).

→ OWASP, Clickjacking Defense Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html

CN

Common Name

Campo de um certificado digital que historicamente identificava o domínio para o qual o certificado foi emitido. Atualmente, os navegadores modernos ignoram o CN e verificam exclusivamente o campo SAN (Subject Alternative Name) para validação do domínio.

→ RFC 5280: datatracker.ietf.org/doc/html/rfc5280

CRIME

Compression Ratio Info-leak Made Easy

Ataque publicado em 2012 que explora a compressão na camada TLS para inferir o conteúdo de cookies de sessão, analisando variações no tamanho dos dados comprimidos. Levou os principais navegadores e servidores a desativarem a compressão TLS. O TLS 1.3 eliminou o suporte à compressão por completo.

→ OWASP, Transport Layer Security Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

→ CVE-2012-4929: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-4929

CRL

Certificate Revocation List — Lista de Revogação de Certificados

Lista publicada por uma Autoridade Certificadora contendo os certificados que foram revogados antes do prazo de validade, por comprometimento da chave privada ou outros motivos. Mecanismo tradicional de verificação de revogação, progressivamente substituído pelo OCSP e pelo OCSP Stapling.

→ RFC 5280: datatracker.ietf.org/doc/html/rfc5280

CSP

Content Security Policy

Cabeçalho HTTP que permite ao administrador do site definir uma lista de origens confiáveis para carregar scripts, estilos, imagens e outros recursos, impedindo que o navegador execute conteúdo de fontes não autorizadas. É uma das principais defesas contra ataques de Cross-Site Scripting (XSS).

→ W3C Content Security Policy Level 3: w3.org/TR/CSP3

→ MDN Web Docs, `Content-Security-Policy`: developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy

→ OWASP, Content Security Policy Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html

D

DANE

DNS-Based Authentication of Named Entities

Protocolo que permite vincular um certificado TLS diretamente ao DNS do domínio, por meio de um registro `TLSA`, reduzindo a dependência exclusiva das Autoridades Certificadoras para validação de identidade. Requer DNSSEC como pré-requisito.

→ RFC 6698 — The DNS-Based Authentication of Named Entities (DANE) Protocol: datatracker.ietf.org/doc/html/rfc6698

→ RFC 7671 — The DANE Protocol: Updates and Operational Guidance: datatracker.ietf.org/doc/html/rfc7671

→ Internet Society, Deploy360, DANE: internetsociety.org/deploy360/dane

DoS e DDoS

Denial of Service e Distributed Denial of Service — Negação de Serviço e Negação de Serviço Distribuída

Ataque que esgota os recursos de um servidor ou serviço — capacidade de processamento, memória ou banda — a ponto de torná-lo indisponível para usuários legítimos. No **DoS**, o ataque parte de uma única origem, podendo explorar operações que consomem recursos de forma desproporcional, como a renegociação TLS iniciada pelo cliente. No **DDoS**, o ataque parte de múltiplas origens simultaneamente, geralmente por meio de uma rede de máquinas comprometidas, o que amplia o volume e dificulta o bloqueio.

→ NIST SP 800-61 Rev. 3, Computer Security Incident Handling Guide: csrc.nist.gov/pubs/sp/800/61/r3/final

DHE

Diffie-Hellman Ephemeral

Variante do algoritmo de troca de chaves Diffie-Hellman que gera uma chave nova e temporária para cada sessão, garantindo a propriedade de sigilo futuro (Forward Secrecy). O tamanho mínimo recomendado para os parâmetros DHE é de 2048 bits.

→ RFC 7919 — Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for TLS: datatracker.ietf.org/doc/html/rfc7919

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

DKIM

DomainKeys Identified Mail

Mecanismo de autenticação de e-mail que adiciona uma assinatura criptográfica ao cabeçalho de cada mensagem enviada, permitindo que o servidor destinatário verifique que o e-mail foi realmente enviado pelo domínio declarado e que seu conteúdo não foi alterado em trânsito.

→ RFC 6376 — DomainKeys Identified Mail (DKIM) Signatures: datatracker.ietf.org/doc/html/rfc6376

DMARC

Domain-Based Message Authentication, Reporting, and Conformance

Protocolo de autenticação de e-mail que define a política a ser seguida pelos servidores destinatários quando um e-mail falha nas verificações de SPF ou DKIM. Também gera relatórios detalhados sobre tentativas de uso indevido do domínio. As políticas disponíveis são: `p=none`, `p=quarantine` e `p=reject`.

→ RFC 7489 — Domain-based Message Authentication, Reporting, and Conformance (DMARC): datatracker.ietf.org/doc/html/rfc7489

→ dmarc.org: dmarc.org

DNS

Domain Name System — Sistema de Nomes de Domínio

Sistema hierárquico e distribuído que traduz nomes de domínio legíveis por humanos (ex.: meusite.gov.br) em endereços IP utilizados para comunicação na internet. É considerado uma infraestrutura crítica da internet.

→ RFC 1034 — Domain Names — Concepts and Facilities: datatracker.ietf.org/doc/html/rfc1034

→ RFC 1035 — Domain Names — Implementation and Specification: datatracker.ietf.org/doc/html/rfc1035

DNSSEC

Domain Name System Security Extensions

Conjunto de extensões de segurança do DNS que adiciona assinaturas criptográficas às respostas DNS, garantindo autenticidade e integridade. Protege contra ataques como envenenamento de cache DNS e DNS Spoofing. Não oferece confidencialidade nem disponibilidade.

→ RFC 4033, 4034, 4035 — DNS Security Introduction and Requirements: datatracker.ietf.org/doc/html/rfc4033

→ RFC 9364 — DNSSEC: datatracker.ietf.org/doc/html/rfc9364

→ NIST SP 800-81 Rev. 3, Secure Domain Name System Deployment Guide: csrc.nist.gov/pubs/sp/800/81/r3/final

→ DNSViz: dnsviz.net

DNS Spoofing

Falsificação de DNS

Ataque em que respostas DNS falsas são inseridas na comunicação entre o cliente e o servidor DNS, redirecionando usuários para endereços IP controlados pelo invasor. O DNSSEC é a principal proteção contra esse tipo de ataque.

→ NIST SP 800-81 Rev. 3: csrc.nist.gov/pubs/sp/800/81/r3/final

Domain Spoofing

Falsificação de Domínio

Prática em que um invasor envia e-mails utilizando o domínio de uma organização legítima como remetente, sem autorização. Mitigado pela combinação de SPF, DKIM e DMARC.

→ NIST SP 1800-6: csrc.nist.gov/pubs/sp/1800/6/final

Downgrade Attack

Ataque de Rebaixamento

Ataque em que um invasor força o servidor e o cliente a utilizarem uma versão mais antiga e menos segura de um protocolo, explorando vulnerabilidades conhecidas dessa versão. Exemplos: forçar o uso de TLS 1.0 em vez de TLS 1.3, ou de uma cifra fraca em vez de uma forte.

→ OWASP, Transport Layer Security Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html

DS Record

Delegation Signer Record

Registro DNS publicado na zona do domínio pai que contém um hash da chave pública KSK do domínio filho, criando o elo de confiança entre as zonas DNS no DNSSEC. É o mecanismo pelo qual a cadeia de confiança DNSSEC é estabelecida do domínio raiz até o domínio específico.

→ RFC 4034 — Resource Records for the DNS Security Extensions: datatracker.ietf.org/doc/html/rfc4034

E

ECDHE

Elliptic Curve Diffie-Hellman Ephemeral

Variante do algoritmo de troca de chaves Diffie-Hellman baseada em curvas elípticas, que gera uma chave nova e temporária para cada sessão. Combina a eficiência das curvas elípticas com a propriedade de sigilo futuro (Forward Secrecy). Algoritmo recomendado pelo NIST e amplamente utilizado no TLS 1.3.

→ RFC 7748 — Elliptic Curves for Security: datatracker.ietf.org/doc/html/rfc7748

→ NIST SP 800-186: csrc.nist.gov/publications/detail/sp/800-186/final

ECDSA

Elliptic Curve Digital Signature Algorithm

Algoritmo de assinatura digital baseado em curvas elípticas, amplamente utilizado em certificados digitais modernos. Oferece segurança equivalente ao RSA com chaves significativamente menores, resultando em melhor desempenho.

→ NIST FIPS 186-5 — Digital Signature Standard: csrc.nist.gov/publications/detail/fips/186/5/final

→ RFC 6090 — Fundamental Elliptic Curve Cryptography Algorithms: datatracker.ietf.org/doc/html/rfc6090

EdDSA

Edwards-Curve Digital Signature Algorithm

Algoritmo de assinatura digital baseado em curvas de Edwards, projetado para oferecer alta segurança e resistência nativa a ataques side-channel. As curvas mais comuns são Ed25519 e Ed448. Progressivamente adotado como alternativa mais segura ao ECDSA.

→ RFC 8032 — Edwards-Curve Digital Signature Algorithm (EdDSA): datatracker.ietf.org/doc/html/rfc8032

Envenenamento de Cache DNS

DNS Cache Poisoning

Ataque em que registros DNS falsos são inseridos no cache de um servidor DNS recursivo, fazendo com que ele retorne respostas incorretas para os usuários, redirecionando-os para endereços IP controlados pelo invasor. Mitigado pelo DNSSEC.

→ NIST SP 800-81 Rev. 3: csrc.nist.gov/pubs/sp/800/81/r3/final

F

FIPS

Federal Information Processing Standards

Padrões de processamento de informação federal publicados pelo NIST para uso em sistemas governamentais dos Estados Unidos. São amplamente adotados como referência internacional de segurança. Exemplos: FIPS 180 (SHA-2), FIPS 202 (SHA-3), FIPS 186-5 (DSS).

→ NIST, FIPS Publications: csrc.nist.gov/publications/fips

Forward Secrecy

Sigilo Futuro — Perfect Forward Secrecy, PFS

Propriedade de um protocolo criptográfico que garante que, mesmo que a chave privada do servidor seja comprometida no futuro, as sessões passadas não possam ser decifradas retroativamente. Alcançada pelo uso de métodos de troca de chaves efêmeros, como ECDHE e DHE.

→ RFC 8446 — The Transport Layer Security (TLS) Protocol Version 1.3: datatracker.ietf.org/doc/html/rfc8446

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

FREAK

Factoring RSA Export Keys

Ataque publicado em 2015 que força o servidor a aceitar chaves RSA de exportação deliberadamente fracas (512 bits), passíveis de fatoração com poder computacional modesto, permitindo ao invasor decifrar a comunicação. Mitigado pela desativação de cifras de exportação e versões antigas do TLS.

→ CVE-2015-0204: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-0204

→ freakattack.com: freakattack.com

Função Hash Criptográfica

Cryptographic Hash Function

Função matemática que transforma uma entrada de tamanho arbitrário em uma saída de tamanho fixo, chamada de hash ou digest. Propriedades essenciais: determinística (mesma entrada sempre produz a mesma saída), unidirecional (impossível reconstruir a entrada a partir do hash) e resistente a colisões (impossível encontrar duas entradas diferentes com o mesmo hash). Exemplos: SHA-256, SHA-384, SHA-512.

→ NIST FIPS 180-4 — Secure Hash Standard: csrc.nist.gov/publications/detail/fips/180/4/final

→ NIST FIPS 202 — SHA-3 Standard: csrc.nist.gov/publications/detail/fips/202/final

G

GCM

Galois/Counter Mode

Modo de operação autenticado de cifras de bloco que combina cifragem e verificação de integridade em uma única operação eficiente. Amplamente utilizado em combinação com o AES (AES-GCM) no TLS 1.3 e recomendado pelo NIST.

→ NIST SP 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM): csrc.nist.gov/publications/detail/sp/800-38d/final

H

Handshake TLS

Processo de negociação inicial entre o cliente e o servidor para estabelecer uma conexão TLS segura. Durante o handshake, as partes negociam a versão do protocolo, os algoritmos criptográficos a serem usados (cifras), trocam certificados e estabelecem as chaves de sessão.

→ RFC 8446 — The Transport Layer Security (TLS) Protocol Version 1.3: datatracker.ietf.org/doc/html/rfc8446

Hash

Resumo Criptográfico

Valor de tamanho fixo produzido por uma função hash criptográfica a partir de uma entrada de tamanho arbitrário. Funciona como uma "impressão digital" dos dados: qualquer alteração na entrada produz um hash completamente diferente.

→ NIST, Glossary: csrc.nist.gov/glossary/term/hash

HSM

Hardware Security Module — Módulo de Segurança de Hardware

Dispositivo físico dedicado ao armazenamento seguro de chaves criptográficas e à realização de operações criptográficas. Protege as chaves contra extração mesmo em caso de comprometimento do sistema operacional. Recomendado para armazenamento das chaves KSK do DNSSEC em infraestruturas críticas.

→ NIST SP 800-57 Part 1 Rev. 5: csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-5/final

HSTS

HTTP Strict Transport Security

Cabeçalho HTTP enviado pelo servidor que instrui o navegador a sempre utilizar HTTPS nas próximas visitas ao site, pelo período definido no parâmetro max-age, eliminando a possibilidade de conexão inicial insegura via HTTP e protegendo contra ataques de SSL Stripping.

→ RFC 6797 — HTTP Strict Transport Security (HSTS): datatracker.ietf.org/doc/html/rfc6797

→ MDN Web Docs, `Strict-Transport-Security`: developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security

HTTP

Hypertext Transfer Protocol

Protocolo de comunicação utilizado para transferência de dados na web, operando na porta `80`. Na ausência de criptografia, os dados trafegam em texto claro, sendo vulneráveis à interceptação e à adulteração.

→ RFC 9110 — HTTP Semantics: datatracker.ietf.org/doc/html/rfc9110

HTTPS

HTTP Secure

Versão segura do HTTP, que combina o protocolo HTTP com uma camada de criptografia fornecida pelo TLS. Garante autenticidade do servidor, confidencialidade e integridade dos dados trafegados. Opera por padrão na porta `443`.

→ RFC 9110 — HTTP Semantics: datatracker.ietf.org/doc/html/rfc9110

→ RFC 8446 — TLS 1.3: datatracker.ietf.org/doc/html/rfc8446

I

IETF

Internet Engineering Task Force

Organização internacional de padronização técnica da internet, responsável pela publicação das RFCs (Request for Comments), documentos que especificam os protocolos e padrões da internet.

→ Site oficial: ietf.org

IPv4

Internet Protocol version 4

Versão 4 do protocolo de endereçamento da internet, que utiliza endereços de 32 bits, permitindo aproximadamente 4,3 bilhões de endereços únicos. O esgotamento dos endereços IPv4 disponíveis é uma realidade consolidada, impulsionando a adoção do IPv6.

→ RFC 791 — Internet Protocol: datatracker.ietf.org/doc/html/rfc791

→ LACNIC, Status do IPv4: lacnic.net/1039/2/lacnic/exhaustion-of-ipv4-addresses

IPv6

Internet Protocol version 6

Versão 6 do protocolo de endereçamento da internet, que utiliza endereços de 128 bits, permitindo um número praticamente ilimitado de endereços únicos. Além de resolver o esgotamento de endereços IPv4, o IPv6 traz melhorias em desempenho, segurança e simplicidade de configuração de rede.

→ RFC 8200 — Internet Protocol, Version 6 (IPv6) Specification: datatracker.ietf.org/doc/html/rfc8200

→ NIST SP 800-119, Guidelines for the Secure Deployment of IPv6: csrc.nist.gov/publications/detail/sp/800-119/final

→ LACNIC, IPv6: lacnic.net/ipv6

K

Key Update

Mecanismo introduzido no TLS 1.3 que permite a atualização das chaves de sessão sem encerrar e reabrir a conexão. Substitui o mecanismo de renegociação do TLS 1.2 de forma mais segura, sem as vulnerabilidades associadas à renegociação insegura.

→ RFC 8446, seção 4.6.3 — Key and IV Update: datatracker.ietf.org/doc/html/rfc8446

KSK

Key Signing Key — Chave de Assinatura de Chaves

Chave criptográfica de longa duração utilizada no DNSSEC para assinar a ZSK (Zone Signing Key). A KSK é referenciada no registro `DS` da zona pai, formando o elo de confiança entre zonas DNS. Por ser de longa duração e alta criticidade, recomenda-se sua rotação anual e armazenamento em HSM.

→ RFC 4033, 4034, 4035: datatracker.ietf.org/doc/html/rfc4033

→ RFC 7583 — DNSSEC Key Rollover Timing Considerations: datatracker.ietf.org/doc/html/rfc7583

L

LACNIC

Latin American and Caribbean Internet Addresses Registry

Registro de recursos de internet responsável pela distribuição e administração dos recursos de numeração IP (IPv4 e IPv6) e de Sistemas Autônomos (AS) na América Latina e Caribe.

→ Site oficial: lacnic.net

Let's Encrypt

Autoridade Certificadora gratuita, automatizada e aberta, operada pela Internet Security Research Group (ISRG), que emite certificados DV (Domain Validation) sem custo. É amplamente utilizada para habilitação do HTTPS em sites de todos os portes.

→ Site oficial: letsencrypt.org

LGPD

Lei Geral de Proteção de Dados Pessoais — Lei nº 13.709/2018

Lei brasileira que estabelece as regras sobre coleta, armazenamento, tratamento e compartilhamento de dados pessoais, impondo obrigações às organizações públicas e privadas que tratam dados de pessoas naturais.

→ Texto da lei: lgpd-brazil.info

→ ANPD: gov.br/anpd

LOGJAM

Ataque publicado em 2015 que força o servidor a aceitar parâmetros Diffie-Hellman de 512 bits (exportação), facilmente fatoráveis, permitindo ao invasor interceptar e decifrar a comunicação. Afeta implementações TLS que suportam cifras DHE_EXPORT.

→ weakdh.org: weakdh.org

→ CVE-2015-4000: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-4000

M

MD5

Message Digest Algorithm 5

Função hash criptográfica publicada em 1991 e amplamente utilizada até meados dos anos 2000. Atualmente considerada quebrada: ataques de colisão práticos foram demonstrados, e certificados digitais assinados com MD5 são rejeitados pelos principais navegadores modernos.

→ RFC 6151 — Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms: datatracker.ietf.org/doc/html/rfc6151

→ NIST SP 800-131A Rev. 2: csrc.nist.gov/publications/detail/sp/800-131a/rev-2/final

MIME Sniffing

Comportamento de alguns navegadores que tentam identificar o tipo de conteúdo de um arquivo analisando seu conteúdo real, mesmo quando o servidor declara um tipo diferente. Pode ser explorado para executar arquivos maliciosos disfarçados de tipos seguros. Mitigado pelo cabeçalho `X-Content-Type-Options: nosniff`.

→ MDN Web Docs, MIME types: developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types

MitM

Man-in-the-Middle — Homem no Meio

Categoria de ataques em que o invasor se posiciona entre o cliente e o servidor, interceptando e potencialmente alterando a comunicação entre as duas partes sem que percebam. Mitigado por criptografia com autenticação forte (HTTPS, TLS, DNSSEC, DANE).

→ NIST SP 800-63B-4 — Digital Identity Guidelines: csrc.nist.gov/pubs/sp/800/63/b/4/final

→ OWASP, Man-in-the-middle attack: owasp.org/www-community/attacks/Manipulator-in-the-middle_attack

MTA-STS

SMTP MTA Strict Transport Security

Mecanismo que permite a um domínio declarar publicamente que a entrega de mensagens aos seus servidores de e-mail exige conexão criptografada com TLS e certificado válido. A declaração é publicada em um registro DNS e em um arquivo de política acessível via HTTPS, o que impede que um invasor suprima a

negociação do STARTTLS para forçar uma conexão em texto claro. Cumpre finalidade semelhante à do DANE para e-mail, com a diferença de não depender de DNSSEC — motivo pelo qual os dois mecanismos são considerados complementares. Não é avaliado pela plataforma TOP.

→ RFC 8461 — SMTP MTA Strict Transport Security (MTA-STS): datatracker.ietf.org/doc/html/rfc8461

MX Record

Mail Exchanger Record

Tipo de registro DNS que especifica os servidores responsáveis por receber e-mails em nome de um domínio, com respectivos valores de prioridade. Quando um servidor quer enviar um e-mail para usuario@dominio.gov.br, consulta os registros `MX` do domínio para saber para qual servidor entregar a mensagem.

→ RFC 1035: datatracker.ietf.org/doc/html/rfc1035

N

NCSC-NL

National Cyber Security Centre — Países Baixos

Centro Nacional de Segurança Cibernética dos Países Baixos, reconhecido internacionalmente pela publicação de guias técnicos de alta qualidade, como o IT Security Guidelines for TLS, amplamente referenciado neste manual.

→ Site oficial: ncsc.nl

→ IT Security Guidelines for TLS v2.1: ncsc.nl/documenten/publicaties/2021/januari/19/ict-beveiligingsrichtlijnen-voor-transport-layer-security-2.1

NIC.br

Núcleo de Informação e Coordenação do Ponto BR

Entidade civil de direito privado, sem fins lucrativos, responsável por implementar as decisões e projetos do CGI.br (Comitê Gestor da Internet no Brasil). Opera o registro de domínios ".br", mantém infraestruturas críticas da internet no Brasil e desenvolve projetos de capacitação e segurança, como a plataforma TOP.

→ Site oficial: nic.br

→ Plataforma TOP: top.nic.br

NIST

National Institute of Standards and Technology

Instituto Nacional de Padrões e Tecnologia dos Estados Unidos, responsável pela publicação de normas e guias técnicos amplamente adotados como referência internacional em segurança da informação, como as séries SP 800, FIPS e NIST CSF.

→ Site oficial: nist.gov

→ Publicações de segurança: csrc.nist.gov/publications

NS Record

Name Server Record

Tipo de registro DNS que indica quais servidores de nomes são autoritativos para um domínio, ou seja, quais servidores respondem pelas consultas DNS daquele domínio.

→ RFC 1035: datatracker.ietf.org/doc/html/rfc1035

O

OCSP

Online Certificate Status Protocol

Protocolo que permite verificar em tempo real se um certificado digital específico foi revogado, consultando diretamente o servidor da Autoridade Certificadora. Alternativa mais eficiente à CRL, mas com implicações de privacidade, pois revela ao servidor da AC quais sites o usuário está acessando.

→ RFC 6960 — X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP): datatracker.ietf.org/doc/html/rfc6960

OCSP Stapling

Mecanismo em que o servidor web busca periodicamente a resposta OCSP junto à Autoridade Certificadora e a inclui diretamente no handshake TLS, eliminando a necessidade de o navegador consultar a AC separadamente. Melhora privacidade e desempenho.

→ RFC 6066, seção 8 — Certificate Status Request: datatracker.ietf.org/doc/html/rfc6066#section-8

OWASP

Open Worldwide Application Security Project

Organização sem fins lucrativos dedicada à melhoria da segurança de software, conhecida pela publicação de guias, ferramentas e listas de referência amplamente adotados, como o OWASP Top 10 e os Cheat

Sheets de segurança.

→ Site oficial: owasp.org

→ OWASP Cheat Sheet Series: cheatsheetseries.owasp.org

P

Phishing

Ataque em que o invasor se passa por uma entidade confiável, como um órgão governamental ou banco, para enganar o usuário e obter informações confidenciais, como senhas e dados pessoais. O e-mail é o principal vetor desse tipo de ataque, mitigado pela autenticação de e-mail (SPF, DKIM e DMARC).

→ NIST SP 800-177 Rev. 1, Trustworthy Email: csrc.nist.gov/publications/detail/sp/800-177/rev-1/final

→ APWG (Anti-Phishing Working Group): apwg.org

PKI

Public Key Infrastructure — Infraestrutura de Chave Pública

Conjunto de políticas, procedimentos, hardware, software e pessoas necessários para criar, gerenciar, distribuir, usar, armazenar e revogar certificados digitais. A PKI é a base do modelo de confiança em que se apoia o HTTPS e a autenticação digital na internet.

→ RFC 5280: datatracker.ietf.org/doc/html/rfc5280

→ Federal PKI — Governança e Infraestrutura (substitui NIST SP 800-32): idmanagement.gov/fpki/

POODLE

Padding Oracle On Downgraded Legacy Encryption

Ataque publicado em 2014 que explora o protocolo SSL 3.0 para decifrar comunicações criptografadas, forçando o servidor a recair para essa versão obsoleta. Mitigado pela desativação completa do SSL 3.0 e versões anteriores do TLS.

→ CVE-2014-3566: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

PSK

Pre-Shared Key — Chave Pré-Compartilhada

Chave criptográfica estabelecida previamente entre cliente e servidor, utilizada no TLS 1.3 para retomada de sessões anteriores sem necessidade de novo handshake completo. Utilizada no mecanismo 0-RTT para

redução de latência em reconexões.

→ RFC 8446, seção 2.2 — Resumption and Pre-Shared Keys: datatracker.ietf.org/doc/html/rfc8446

R

RACCOON Attack

Ataque de temporização publicado em 2020 que explora o protocolo de troca de chaves Diffie-Hellman no TLS 1.2 e versões anteriores. Embora difícil de executar na prática, demonstra a importância da adoção do TLS 1.3, que elimina essa classe de vulnerabilidades.

→ raccoon-attack.com: raccoon-attack.com

→ CVE-2020-1968: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-1968

RC4

Rivest Cipher 4

Algoritmo de cifragem de fluxo, amplamente utilizado no passado em implementações SSL/TLS. Atualmente considerado inseguro, com múltiplas vulnerabilidades documentadas que permitem a recuperação de dados cifrados. Proibido no TLS pela RFC 7465.

→ RFC 7465 — Prohibiting RC4 Cipher Suites: datatracker.ietf.org/doc/html/rfc7465

Referrer

Cabeçalho HTTP Referer

Cabeçalho HTTP enviado pelo navegador ao site de destino informando a URL completa da página de origem quando o usuário clica em um link. Pode conter informações sensíveis presentes na URL, como tokens de autenticação. Controlado pelo cabeçalho `Referrer-Policy`.

→ RFC 9110 — HTTP Semantics, seção 10.1.3: datatracker.ietf.org/doc/html/rfc9110#section-10.1.3

→ W3C Referrer Policy: w3.org/TR/referrer-policy

Registro.br

Departamento do NIC.br responsável pelas atividades de registro e manutenção de nomes de domínio sob o domínio de primeiro nível ".br". Oferece suporte ao DNSSEC sem custo adicional para todos os domínios registrados.

→ Site oficial: registro.br

→ DNSSEC no registro.br: registro.br/tecnologia/dnssec

Renegociação TLS

TLS Renegotiation

Mecanismo do TLS 1.0 ao 1.2 que permite a renegociação dos parâmetros criptográficos durante uma conexão já estabelecida. A ausência de vínculo criptográfico entre o handshake original e o renegociado nas versões antigas criou a vulnerabilidade CVE-2009-3555, corrigida pela RFC 5746. O TLS 1.3 eliminou completamente o mecanismo de renegociação, substituindo-o pelo Key Update.

→ RFC 5746 — Transport Layer Security (TLS) Renegotiation Indication Extension: datatracker.ietf.org/doc/html/rfc5746

→ CVE-2009-3555: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3555

Replay Attack

Ataque de Repetição

Ataque em que o invasor captura pacotes de uma comunicação legítima e os retransmite posteriormente para o servidor, que pode processá-los como se fossem novas requisições autênticas. Especialmente relevante no contexto do modo 0-RTT do TLS 1.3.

→ RFC 8446, seção 8 — 0-RTT Data: datatracker.ietf.org/doc/html/rfc8446

→ RFC 8470 — Using Early Data in HTTP: datatracker.ietf.org/doc/html/rfc8470

RFC

Request for Comments

Documentos técnicos publicados pela IETF que especificam protocolos, procedimentos e convenções da internet. Apesar do nome sugestivo de "pedido de comentários", as RFCs aprovadas constituem os padrões técnicos oficiais da internet.

→ Repositório de RFCs: rfc-editor.org

→ Repositório IETF: datatracker.ietf.org

RSA

Rivest–Shamir–Adleman

Algoritmo de criptografia assimétrica baseado na dificuldade matemática de fatorar o produto de dois números primos grandes. Amplamente utilizado em certificados digitais para assinatura e, historicamente, para troca de chaves. Para uso em certificados, o tamanho mínimo recomendado é de 2048 bits, sendo 3072 bits preferível para certificados com validade superior a dois anos. Chaves de 4096 bits acrescentam custo de processamento sem ganho prático proporcional; quando se busca maior margem de segurança, a migração para ECDSA é preferível.

→ NIST FIPS 186-5: csrc.nist.gov/publications/detail/fips/186/5/final

→ RFC 8017 — PKCS #1: RSA Cryptography Specifications Version 2.2: datatracker.ietf.org/doc/html/rfc8017

S

SAN

Subject Alternative Name

Campo de um certificado digital X.509 que lista todos os domínios, subdomínios e endereços IP para os quais o certificado é válido. Navegadores modernos verificam exclusivamente o campo SAN para validar a correspondência entre o domínio acessado e o certificado apresentado, ignorando o campo CN (Common Name).

→ RFC 5280: datatracker.ietf.org/doc/html/rfc5280

SHA-1

Secure Hash Algorithm 1

Função hash criptográfica de 160 bits desenvolvida pela NSA e publicada pelo NIST em 1995. Considerada insegura desde 2017, quando o ataque SHAttered demonstrou a viabilidade prática de colisões. Certificados assinados com SHA-1 são rejeitados pelos principais navegadores modernos.

→ RFC 6194 — Security Considerations for the SHA-0 and SHA-1 Message-Digest Algorithms: datatracker.ietf.org/doc/html/rfc6194

→ SHAttered: marc-stevens.nl/research/shattered.io/

→ NIST SP 800-131A Rev. 2: csrc.nist.gov/publications/detail/sp/800-131a/rev-2/final

SHA-2

Secure Hash Algorithm 2

Família de funções hash criptográficas publicada pelo NIST, incluindo SHA-224, SHA-256, SHA-384 e SHA-512. SHA-256 é o mínimo recomendado atualmente para assinatura de certificados digitais e operações criptográficas gerais.

→ NIST FIPS 180-4 — Secure Hash Standard: csrc.nist.gov/publications/detail/fips/180/4/final

SHA-3

Secure Hash Algorithm 3

Família de funções hash criptográficas publicada pelo NIST em 2015 (FIPS 202), baseada na função Keccak. Foi desenvolvida como alternativa independente ao SHA-2, com arquitetura interna completamente diferente, aumentando a resiliência caso vulnerabilidades sejam encontradas no SHA-2.

→ NIST FIPS 202 — SHA-3 Standard: csrc.nist.gov/publications/detail/fips/202/final

Side-Channel Attack

Ataque de Canal Lateral

Categoria de ataques que não ataca diretamente a matemática da criptografia, mas explora informações indiretas geradas pelo processamento criptográfico, como tempo de execução (timing attack), consumo de energia (power analysis) ou padrões de acesso à memória cache (cache attack).

→ NIST SP 800-186, seção de segurança de curvas elípticas: csrc.nist.gov/publications/detail/sp/800-186/final

Single-Use Ticket

Mecanismo definido na RFC 8446 que invalida um ticket de sessão TLS após seu primeiro uso, impedindo ataques de Replay no modo 0-RTT. Quando implementado, cada ticket só pode ser utilizado uma única vez para retomada de sessão.

→ RFC 8446, seção 8.1 — Anti-Replay: datatracker.ietf.org/doc/html/rfc8446

SMTP

Simple Mail Transfer Protocol

Protocolo padrão para envio e retransmissão de e-mails entre servidores. Opera na porta `25` para comunicação entre servidores (MTA) e, com suporte a STARTTLS, pode elevar a conexão para criptografia TLS.

→ RFC 5321 — Simple Mail Transfer Protocol: datatracker.ietf.org/doc/html/rfc5321

SOA Record

Start of Authority Record

Tipo de registro DNS que contém informações administrativas sobre a zona DNS, incluindo o servidor de nomes primário, o e-mail do administrador, parâmetros de atualização e o número de série da zona. No contexto do DNSSEC, é o registro cuja assinatura é verificada para confirmar que a zona está corretamente protegida.

→ RFC 1035: datatracker.ietf.org/doc/html/rfc1035

SPF

Sender Policy Framework

Mecanismo de autenticação de e-mail que permite ao proprietário de um domínio publicar no DNS uma lista de servidores e endereços IP autorizados a enviar e-mails em nome daquele domínio. Os servidores destinatários podem verificar essa lista para identificar e-mails enviados por remetentes não autorizados.

→ RFC 7208 — Sender Policy Framework (SPF) for Authorizing Use of Domains in Email: datatracker.ietf.org/doc/html/rfc7208

SSL

Secure Sockets Layer

Protocolo predecessor do TLS, desenvolvido pela Netscape nos anos 1990. Todas as versões do SSL (2.0 e 3.0) estão completamente descontinuadas e não devem mais ser utilizadas, pois contêm vulnerabilidades graves e publicamente exploráveis, como o POODLE.

→ RFC 7568 — Deprecating Secure Sockets Layer Version 3.0: datatracker.ietf.org/doc/html/rfc7568

SSL Labs

Ferramenta gratuita de análise de configuração TLS desenvolvida pela Qualys, amplamente utilizada para avaliar a segurança de servidores HTTPS, verificar versões de protocolo suportadas, cifras configuradas, qualidade da cadeia de certificados e conformidade com boas práticas.

→ SSL Labs: ssllabs.com/ssltest

SSL Stripping

Ataque em que o invasor intercepta a conexão HTTP inicial antes do redirecionamento para HTTPS e mantém o usuário em uma conexão insegura, enquanto ele mesmo se comunica com o servidor via HTTPS. O usuário acredita estar acessando o site normalmente, mas sua comunicação não está criptografada. Mitigado pelo HSTS.

→ OWASP, Manipulator-in-the-Middle attack (inclui SSL Stripping): owasp.org/www-community/attacks/Manipulator-in-the-middle_attack

STARTTLS

Extensão de protocolo que permite elevar uma conexão inicialmente insegura para uma conexão criptografada via TLS, utilizando o mesmo canal de comunicação já estabelecido. Amplamente utilizado em SMTP (e-mail) e outros protocolos de mensagens. Diferente do TLS implícito, o STARTTLS inicia a comunicação sem criptografia e negocia o TLS posteriormente.

→ RFC 3207 — SMTP Service Extension for Secure SMTP over Transport Layer Security: datatracker.ietf.org/doc/html/rfc3207

→ RFC 8314 — Cleartext Considered Obsolete: datatracker.ietf.org/doc/html/rfc8314

SWEET32

Ataque publicado em 2016 que explora a limitação do tamanho de bloco de 64 bits em cifras como 3DES e Blowfish. Capturando grandes volumes de tráfego criptografado com essas cifras, o invasor pode recuperar dados em texto claro por força estatística. Mitigado pela desativação de cifras com blocos de 64 bits.

→ sweet32.info: sweet32.info

→ CVE-2016-2183: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-2183

T

TLS

Transport Layer Security

Protocolo criptográfico que garante autenticidade, confidencialidade e integridade na comunicação entre sistemas na internet. Sucessor do SSL, é a base do HTTPS. A versão atual e recomendada é o TLS 1.3, definida na RFC 8446.

→ RFC 8446 — The Transport Layer Security (TLS) Protocol Version 1.3: datatracker.ietf.org/doc/html/rfc8446

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/pubs/sp/800/52/r2/final

TLSA Record

Tipo de registro DNS utilizado pelo DANE para publicar o hash do certificado TLS de um servidor, vinculando-o diretamente ao DNS do domínio. O registro `TLSA` especifica o protocolo, a porta e o servidor para os quais o certificado é válido, além do hash criptográfico do certificado ou de sua chave pública.

→ RFC 6698: datatracker.ietf.org/doc/html/rfc6698

→ RFC 7671: datatracker.ietf.org/doc/html/rfc7671

TOP

Teste os Padrões

Plataforma gratuita do NIC.br que realiza testes automatizados de segurança e boas práticas em domínios ".br", avaliando segurança web (HTTPS, TLS, cabeçalhos de segurança, DNSSEC), autenticação de e-mail (SPF, DKIM, DMARC, DANE, STARTTLS) e modernização de endereçamento (IPv6).

→ Plataforma TOP: top.nic.br

TXT Record

Text Record

Tipo de registro DNS utilizado para publicar informações textuais arbitrárias associadas a um domínio. É o formato utilizado para publicar registros SPF, registros DMARC e chaves públicas DKIM no DNS do domínio.

→ RFC 1035: datatracker.ietf.org/doc/html/rfc1035

W

Wildcard Certificate

Certificado Coringa

Certificado digital que cobre um domínio e todos os seus subdomínios de primeiro nível, utilizando o asterisco como curinga no campo SAN (ex.: *.dominio.gov.br cobre www.dominio.gov.br, sistemas.dominio.gov.br, etc., mas não sub.sub.dominio.gov.br). Simplifica a gestão de certificados em ambientes com múltiplos subdomínios.

→ RFC 5280: datatracker.ietf.org/doc/html/rfc5280

X

X.509

Padrão internacional que define o formato dos certificados digitais utilizados em PKI, HTTPS, S/MIME e outros protocolos de segurança. Especifica a estrutura dos certificados, incluindo campos como versão, número de série, algoritmo de assinatura, emissor, validade, sujeito, chave pública, SAN e extensões.

→ RFC 5280 — Internet X.509 PKI Certificate and CRL Profile: datatracker.ietf.org/doc/html/rfc5280

XSS

Cross-Site Scripting

Categoria de vulnerabilidades em aplicações web em que um invasor consegue injetar scripts maliciosos em páginas visualizadas por outros usuários. Os scripts são executados no navegador da vítima, podendo roubar cookies de sessão, credenciais e dados pessoais. Mitigado por cabeçalhos como `Content-Security-Policy` e `X-Content-Type-Options`.

→ OWASP, Cross Site Scripting (XSS): owasp.org/www-community/attacks/xss

→ OWASP, XSS Prevention Cheat Sheet: cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html

Z

0-RTT

Zero Round Trip Time

Recurso do TLS 1.3 que permite ao cliente enviar dados para o servidor antes mesmo de o handshake ser concluído, utilizando uma chave pré-compartilhada de uma sessão anterior. Reduz a latência em reconexões, mas introduz vulnerabilidade a ataques de repetição (Replay Attack). O NIST recomenda que, em geral, servidores não devem suportar 0-RTT.

→ RFC 8446, seção 2.3 — 0-RTT Data: datatracker.ietf.org/doc/html/rfc8446

→ RFC 8470 — Using Early Data in HTTP: datatracker.ietf.org/doc/html/rfc8470

→ NIST SP 800-52 Rev. 2: csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

ZSK

Zone Signing Key — Chave de Assinatura de Zona

Chave criptográfica utilizada no DNSSEC para assinar os registros DNS da zona. É referenciada pela KSK e deve ser rotacionada com maior frequência, recomendando-se intervalos de 30 a 90 dias, para limitar o impacto de um eventual comprometimento.

→ RFC 4033, 4034, 4035: datatracker.ietf.org/doc/html/rfc4033

→ RFC 7583 — DNSSEC Key Rollover Timing Considerations: datatracker.ietf.org/doc/html/rfc7583