Projeto de estrutura de um servidor web implementando normas e protocolos de segurança segundo o projeto *Open Standards Everywhere*

Raphaela Silva Goulart[1], **Silvia Calmon de Albuquerque**[1]

[1] Departamento de Computação – Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)
Avenida Amazonas, 7675 – 30.510-000 – Belo Horizonte – MG – Brasil

raphaela.sgoulart@gmail.com, silviacalmon@cefetmg.br

Abstract. *The Open Standards Everywhere project was created in order to help web servers have high availability and be as secure as possible, using standards and protocols defined by the Internet Engineering Task Force. The objective of this work was to analyze whether some web servers are respecting these standards and to indicate improvements that can be made. In addition, to propose a structure project for a web server that implements these standards and protocols, using free resources and services. Through this analysis, it was evident the low adherence rate of the protocols by popular sites from different branches, showing even more the importance of the server structure project, facilitating the creation of more secure web servers.*

Resumo. *O projeto Open Standards Everywhere foi criado com o intuito de ajudar que servidores web tenham alta disponibilidade e sejam o mais seguro possível, a partir do uso de normas e protocolos definidos pela Internet Engineering Task Force. O objetivo desse trabalho foi analisar se alguns servidores web estão respeitando essas normas e indicar melhorias que podem ser feitas. Além disso, propor um projeto de estrutura de um servidor web que implementa essas normas e protocolos, utilizando recursos e serviços gratuitos. Por meio dessa análise, ficou evidente a baixa taxa de adesão dos protocolos por sites populares de diferentes ramos, mostrando ainda mais a importância do projeto de estrutura do servidor, facilitando a criação de servidores web mais seguros.*

1. Introdução

A web se mostra onipresente fazendo parte do comércio, entretenimento e interação social atualmente. De acordo com dados da *International Telecommunication Union* (ITU), 2019 foi o primeiro ano em que mais da metade do mundo (51,2%, ou 3,9 bilhões de pessoas) participou do mundo digital [BROADBAND COMMISSION 2019], para o qual a migração traz riscos e problemas de segurança da informação.

A *Internet Society* (ISOC) é uma organização global dedicada a garantir que a Internet permaneça aberta, transparente e definida por todos. Seus esforços visam aumentar o alcance e a confiabilidade da Internet a curto prazo, bem como garantir os alicerces para um crescimento contínuo e sólido [ISOC 2020a]. Um de seus projetos é o *Open Standards Everywhere* (OSE)[1] que tem como intuito incentivar que servidores web sejam

[1] https://www.internetsociety.org/issues/open-standards-everywhere/

mais seguros e disponíveis. Para isso, foi compilada uma série de diretrizes para melhorar a confiabilidade dos serviços *online*, baseada nas normas e protocolos definidos pela *Internet Engineering Task Force* (IETF), uma comunidade internacional cuja missão é a padronização dos protocolos da Internet [IETF 2020].

O objetivo desse trabalho foi realizar um estudo de caso para analisar se servidores web de alguns sites populares estão respeitando essas diretrizes e para indicar melhorias a serem feitas para alcançar esse padrão. Além disso, propôs-se um projeto de estrutura de um servidor web que implementa as normas e os protocolos indicados pelo projeto OSE.

Pôde-se comprovar a baixa taxa de uso dos protocolos seguros em sites de diferentes ramos muito acessados no Brasil. A partir disso, reforça-se a importância do projeto de estrutura do servidor, uma vez que este demonstra a criação de servidores web mais seguros, nos quais as normas e protocolos indicados pela ISOC estão implementados. Além disso, esse trabalho demonstra que existem recursos gratuitos e de fácil utilização que permitem construir um servidor mais seguro, indicando que a falta de uso desses protocolos deve-se aparentemente por desconhecimento por parte dos administradores de servidores web e não por dificuldades na implementação das normas ou por falta de recursos.

2. Referencial Teórico

Cada vez mais os negócios estão crescendo na web e, com isso, uma grande quantidade de dados e de processos confidenciais e sensíveis está se tornando vulnerável. O projeto *Open Web Application Security Project* (OWASP) *Top Ten* de 2017 apontou os seguintes dez maiores riscos em aplicações web [OWASP 2017]:

1. Injeção
2. Quebra de autenticação
3. Exposição de dados sensíveis
4. Entidades Externas de XML (XXE)
5. Quebra de controle de acesso
6. Configurações de segurança incorretas
7. *Cross-Site Scripting* (XSS)
8. Desserialização insegura
9. Uso de componentes vulneráveis
10. Registro e monitorização insuficientes

A maioria desses riscos ocorrem por causa do uso de tecnologias, configurações e recursos antigos, sendo que já existem soluções disponíveis para evitá-los ou, pelo menos, para reduzí-los, como a aplicação das diretrizes propostas pelo projeto OSE para os servidores web [ISOC 2020b], como os protocolos e as opções de segurança indicados a seguir:

- IPv6 (versão mais atual do protocolo IP) [Deering and Hinden 1998]
- DNSSEC (versão segura do protocolo DNS) [Arends et al. 2005]
- Conexão Segura
  - HTTPS [Rescorla 2000]
  - TLS (versão 1.3) [Rescorla 2018]
  - Criptografia segura, parâmetros de ordem de preferência de métodos e de troca de chaves [BROADBAND COMMISSION 2019]
  - Certificado digital [Durumeric et al. 2013]
  - DANE [Dukhovni and Hardaker 2015]
- Opções de Segurança: *X-Frame*, *X-Content-Type*, *X-XSS-Protection*, CSP, etc.

3. Trabalhos Relacionados

Existem alguns estudos voltados para a análise de segurança em sites, verificando a existência de parte dos protocolos abordados nesse trabalho. O estado da segurança HTTP em sites de banco foi analisada por Aditya Sood e Richard Enbody (2011), testando a aceitabilidade de novos recursos de proteção em um dos setores mais visados, uma vez que a exploração de vulnerabilidades no site de um banco pode expor transações monetárias *online* para fraudes. O objetivo desse estudo era verificar se as páginas críticas (páginas de configuração e com autenticação) nos sites dos bancos estavam implementando segurança declarativa, ou seja, utilizando cabeçalhos HTTP com alvo em vetores de ataque específicos.

Ainda analisando a segurança de sites, um estudo sobre os cabeçalhos de segurança dos 1.000.000 sites melhores posicionados, de acordo com o ranking da Alexa, foi feito por Artūrs Lavrenovs e F. Jesús Rubio Melón (2018). Eles apresentaram um experimento que consistia em, para cada domínio analisado, enviar requisições HTTP/1.1 e HTTPS para o próprio domínio e o subdomínio "www" correspondente. Na resposta obtida era verificada a presença dos cabeçalhos de segurança analisados *Strict-Transport-Security*, *Content-Security-Policy*, *X-XSS-Protection*, *X-Frame-Options*, *Set-Cookie* e *X-Content-Type*, e analisados os cabeçalhos que podem carregar informações sobre o servidor.

4. Metodologia

Esse trabalho foi desenvolvido em quatro etapas. Inicialmente, fêz-se o levantamento de todas as normas e protocolos de segurança indicados pelo projeto OSE, apresentados na Seção 2. Essa etapa foi a base do trabalho, pois auxiliou no entendimento de quais são as fragilidades de um sistema caso certos protocolos não sejam implementados, além de ser fundamental para a construção do projeto do servidor web seguro.

Em seguida, foi realizada uma análise de sites de domínios já existentes e muito acessados no Brasil e no mundo, segundo a Alexa[2]. A partir dessa análise, descrita na Seção 5, foi possível ter um panorama geral do nível de segurança dos sites, além de mostrar quais os protocolos indicados pelos projeto OSE são mais ou menos utilizados.

As últimas etapas foram a construção, descrita na Seção 6, e os testes, descritos na Seção 7, do servidor web, segundo as diretrizes do projeto OSE.

5. Análise de sites

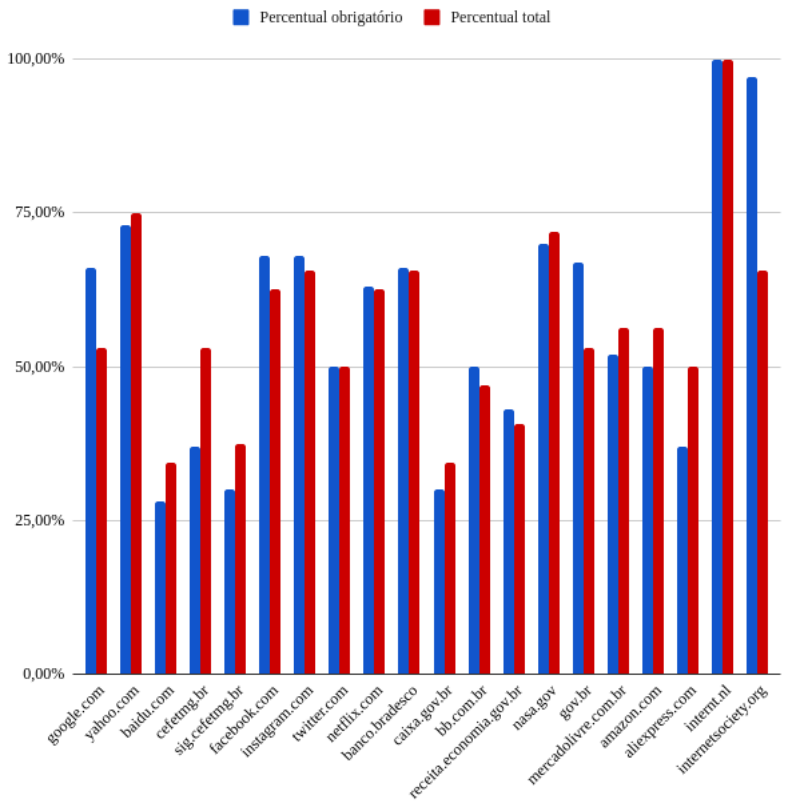
A primeira análise desse trabalho consistiu em verificar a implementação das 32 normas e protocolos propostos pela ISOC, apresentados na Tabela 1, dentre 20 domínios de diferentes ramos: 3 sites de ferramentas de busca, 3 sites de redes sociais, 3 sites de bancos, 3 sites de *e-commerce* e 1 site de *streaming*, que possuem um grande número de acesso, estando entre os 500 sites mais acessados do Brasil ou do mundo, segundo a Alexa. Foram escolhidos também 2 sites do governo brasileiro, não considerados seguros, e 1 site do governo dos Estados Unidos, considerado altamente seguro [Ribeiro 2019]. Por fim, foram escolhidos 2 sites utilizados na instituição de ensino CEFET-MG , o site da ISOC, que possui o projeto OSE, e o próprio site do teste.

[2]https://www.alexa.com/

Nesse processo, foi utilizada a ferramenta de teste criada pela *Internet.nl*, uma iniciativa da *Dutch Internet Standards Platform*, que tem como objetivo verificar o uso de padrões modernos da Internet. Após o usuário fornecer o nome de domínio de um site, é verificado se este possui suporte para IPv6, DNSSEC, HTTPS e outras opções de segurança, sendo que uma pontuação de 100% significa que um site está em conformidade com os padrões de teste propostos. A pontuação final obtida considera apenas as categorias consideradas como obrigatórias, não pontuando as que são recomendadas ou opcionais.

Para os 20 domínios selecionados, foram analisados o percentual de protocolos obrigatórios implementados por cada site e o percentual total, dentre todos os protocolos testados, obrigatórios ou não. Por meio dos resultados exibidos no gráfico comparativo da Figura 1, pode-se notar uma baixa adesão dos protocolos por sites muito acessados e, principalmente, sites que lidam com dados confidenciais, como sites bancários. Analisando os percentuais alcançados, percebe-se que o único site, dentre os analisados, que implementa 100% dos protocolos é o próprio site que realiza os testes, o *internet.nl*. Dentre os sites de bancos analisados, nenhum deles obteve uma pontuação maior que 70%, sendo o site da Caixa Econômica Federal o que possui a menor taxa de implementação, com apenas 30% dos obrigatórios e 34,38% do total. O site que obteve a menor pontuação foi o site de busca *baidu.com*, obtendo apenas 28% dentre os obrigatórios e 34,38% do total. Analisando os outros ramos, nenhum dos sites de compra *online* obteve uma pontuação maior que 60%. Entre os sites de redes sociais, todos obtiveram uma taxa de implementação entre 50% e 70%.

Figura 1. Comparação dos sites



O percentual de implementação de cada protocolo também foi analisado. Por meio da análise dos resultados obtidos na Tabela 1, observa-se a implementação de forma individual e, a partir disso, é possível ter uma visão de quais protocolos são mais implementados e quais são mais negligenciados. Dentre os protocolos considerados obrigatórios na análise, o que possui menor percentual, sendo que apenas 5% dos sites o implementaram, é a imposição da preferência da criptografia, que verifica se o servidor web impõe sua própria preferência de cifra ao negociar com um navegador e oferece cifras preferencialmente seguras e rápidas. Para a escolha do método, é recomendado utilizar as que realizam a troca de chaves com base em curvas elípticas ao invés das que usam campos finitos, sendo ambas preferidas às que usam uma troca de chave estática. Também é recomendado utilizar cifras que fazem criptografia em massa com base nos algoritmos *Authenticated Encryption with Associated Data* (AEAD) e que fazem a verificação de certificado com base no *Elliptic Curve Digital Signature Algorithm* (ECDSA). A ausência dessa imposição possibilita a um invasor fazer uso de uma conexão SSL/TLS insegura, uma vez que o cliente pode utilizar formas menos eficientes de criptografia.

Ainda entre os protocolos considerados obrigatórios, o DNSSEC alcançou apenas 15% de implementação, sendo este de grande importância. Sua ausência facilita ataques como DNS *cache-poisoning*, que consiste em inserir informações falsas em um cache DNS, para que as consultas retornem uma resposta incorreta e os usuários sejam direcionados aos sites errados [Cloudfare 2020], comprometendo a segurança ou a integridade dos dados.

Todos os sites analisados implementaram o HTTPS, mas 20% desses sites não implementaram o redirecionamento de HTTP para HTTPS, dessa forma, eles continuam acessíveis em um domínio sem essa opção de segurança. Analisando os protocolos relacionados aos certificados, 100% dos sites possuem uma cadeia de confiança do certificado completa e assinada por uma autoridade de certificação raiz confiável, com assinatura digital do certificado utilizando parâmetros seguros, assinado com um algoritmo *hash* seguro e com o nome de domínio presente no certificado correspondente ao nome de domínio do site.

Tabela 1. Análise dos protocolos

Protocolos		Percentual
IPv6	Nome de servidores com IPv6	75%
	Servidores acessíveis via IPv6	75%
	Servidores web com endereço IPv6	45%
	Servidores web acessíveis via IPv6	45%
	Mesmo website alcançado via IPv4 e IPv6	45%
DNSSEC	Existência	15%
	Validação	15%
Conexão Segura	HTTPS	100%
	Redirecionamento de HTTP para HTTPS	80%
	Compressão HTTP	50%
	Política HSTS	65%
	Versão de TLS	35%
	Criptografia segura	20%
	Imposição da preferência da criptografia	5%
	Parâmetros seguros para troca de chaves	75%
	Função de hash segura para troca de chaves	100%
	Compressão TLS	100%
	Renegociação segura	100%
	Bloqueio de renegociação iniciada pelo cliente	60%
	0-RTT	100%
	OCSP	40%
	Cadeia de certificado	100%
	Certificado de chave pública	100%
	Assinatura do certificado	100%
	Nome do domínio no certificado	100%
	Existência do DANE	5%
	Validação do DANE	5%
Opções de segurança	X-Frame-Options	50%
	X-Content-Type-Options	45%
	X-XSS-Protection	30%
	CSP	20%
	Referrer-Policy	15%

6. Desenvolvimento do servidor

Após o levantamento e o estudo dos protocolos necessários, o próximo passo foi realizar a implementação do servidor web de acordo com as normas e protocolos definidos pelo OSE. As seções a seguir apresentam as etapas realizadas nesse desenvolvimento.

6.1. Configuração inicial do servidor

Primeiramente, foi necessário instalar o servidor que, para esse projeto, foi escolhido o Nginx, um servidor HTTP *open source* mais leve e que consome menos memória que o tradicional Apache [Jankov 2020]. O projeto OSE contém documentação direcionada a esses dois tipos de servidores web *open source*.

O próximo passo foi registrar um nome de domínio para esse servidor para que ele possuísse acesso mais fácil pela Internet. Para isso foi escolhido o deSEC[3], um serviço de hospedagem DNS gratuito, projetado com diversas opções de segurança. No site da organização, é possível realizar o cadastro com o e-mail e obter o nome do domínio escolhido se este estiver disponível.

Após isso, foi necessário colocar os endereços IP para os quais esse nome de domínio aponta. Primeiramente, foi criado o registro A, que contém o IPv4 desejado. Em seguida, foi necessário cadastrar o registro AAAA que contém o IPv6.

6.2. Certificados

Com um nome de domínio já definido e configurado, a próxima etapa foi a criação dos certificados. Esse passo deve ocorrer apenas após a configuração do nome de domínio, pois uma das condições para se obter uma conexão segura, segundo o projeto OSE, é que este nome esteja presente no certificado da página. A CA escolhida para emitir o certificado foi a Let's Encrypt, uma autoridade de certificação que fornece gratuitamente certificados de criptografia TLS X.509 através de um processo automatizado. Para a criação dos certificados foi utilizado o Certbot, uma ferramenta de software que gera certificados emitidos pela Let's Encrypt.

O deSEC, utilizado na hospedagem DNS do servidor, oferece um *script* (*hook.sh*) que automatiza o processo de geração do certificado e autenticação do domínio, necessária para utilização do DNSSEC, além de uma *Application Programming Interface* (API) com diversos serviços que são utilizados pelo Certbot na geração de certificados. Através dessas ferramentas é possível gerar o certificado adequado para seguir com configuração do servidor.

6.3. DANE

Para gerar o registro TLSA, a fim de se utilizar o DANE, foi usada uma ferramenta fornecida por Shumon Huque[4]. Essa ferramenta gera um registro TLSA a partir de um certificado e dos parâmetros fornecidos.

O primeiro parâmetro refere-se ao uso do certificado, que especifica a associação que será usada para equiparar o certificado apresentado no *handshake* do TLS. O segundo parâmetro é o de seleção, que especifica qual parte do certificado TLS apresentado pelo servidor será comparado com os dados de associação. O certificado gerado pelo Certbot utiliza o SHA-256, dessa forma, foi utilizado o mesmo tipo para geração do TLSA. Em seguida, é necessário copiar o conteúdo do arquivo fullchain.pem gerado pelo Certbot. Por fim, insere-se o número da porta, 443 padrão do HTTPS, o protocolo de transporte e o nome de domínio.

O registro TLSA gerado foi inserido no servidor de DNS, neste caso o deSEC. O subdomínio deve ser inserido no formato mostrado no momento da sua geração. Os registros finais usados no servidor de DNS são o registro TLSA, para a utilização do DANE, registro tipo AAAA (IPv6), registro tipo A (IPv4) e registro NS, que indica qual servidor de DNS é autoritativo para o domínio em questão, sendo este inicializado pelo próprio deSEC.

[3]https://desec.io/

[4]https://www.huque.com/

6.4. Inclusão dos protocolos

A última etapa do desenvolvimento consistiu em alterar a configuração do Nginx, para a utilização dos certificados instalados e dos protocolos definidos pela OSE. No arquivo de configuração utilizado pelo servidor, foi necessário incluir alguns itens, como pode ser obtido no GitHub https://github.com/RaphaGoulart/Servidor-WEB-OSE.

6.5. Configuração do roteador

No caso desse trabalho, o servidor web foi instalado em uma rede doméstica, por isso, foi necessária a configuração do roteador para permitir o acesso externo ao servidor e um redirecionamento de porta, tendo em vista que as portas web padrão, 80 e 443, normalmente são bloqueadas pelas operadoras para clientes pessoa física. O uso do protocolo IPv6 também foi habilitado para cumprir as orientações do projeto OSE. Para isso, foi preciso habilitar seu uso, configurar sua conexão com a mesma sessão *Point-to-Point Protocol over Ethernet* (PPPoE) utilizada pelo roteador para o IPv4 e usar o tipo de endereçamento *Stateless Address Autoconfiguration* (SLAAC). Este tipo de endereçamento utiliza um método para gerar identificadores de interface IPv6 de modo que este seja estável em cada sub-rede, mas o identificador de interface correspondente muda quando o *host* muda de uma rede para outra [Gont 2014].

7. Avaliação do servidor

Após o desenvolvimento do servidor foi feita sua avaliação, analisando a implementação dos protocolos definidos pela OSE. Essa avaliação está descrita nas seções a seguir.

7.1. IPv6

O primeiro teste realizado foi o de conectividade IPv6. Para isso, foi utilizado o site IPv6 - *ARE YOU CONNECTED*?[5], criado para ajudar a aumentar a conscientização sobre erros comuns de configuração IPv6 em geral.

Este site realiza uma série de testes. Primeiramente, ele verifica a conectividade dos servidores DNS, que no caso do deSEC, possuem conexão IPv6. Em seguida, ele testa se o DNS possui conectividade com usuários *IPv6-only*, ou seja, aqueles que utilizam apenas o IPv6, sem possuirem o IPv4, sendo este teste também positivo. O próximo teste realizado foi o do registro de IPv6 (AAAA) para o *Mail exchangers* (MX), que indica como as mensagens de e-mail devem ser roteadas de acordo com o *Simple Mail Transfer Protocol* (SMTP), o protocolo padrão para todos os e-mails [Cloudfare 2021]. Como o objetivo desse trabalho era implementar um servidor web e não um servidor de e-mail, nenhum registro MX foi utilizado. Por fim, ele testa dois nomes de domínio, com e sem o www inicial. O servidor implementado não possui um subdomínio de nome www, dessa forma, nenhum registro faz referência a ele. Ao final do teste, o servidor obteve nota 5 de um total de 5, mostrando sua acessibilidade via IPv6.

7.2. Certificados

Um dos pontos a ser analisado é a utilização de certificados com parâmetros considerados seguros. É possível ver algumas informações do certificado de um site através do

[5]https://ip6.nl/

visualizador de informações, que o próprio navegador possui, clicando no cadeado que aparece ao lado esquerdo da URL, quando o site possui certificado. Um dado importante é a assinatura digital, a usada nesse projeto utiliza o SHA-256, um algoritmo *hash* dentre a lista dos considerados adequados segundo o Centro Nacional de *Cyber* Segurança da Holanda (NCSC-NL) [NATIONAL CYBER SECURITY CENTRE 2019].

A fim de se realizar uma análise mais completa e detalhada, foram utilizadas outras ferramentas de algumas companhias, entre elas a Geocerts, uma empresa privada que comercializa, distribui e oferece suporte a certificados SSL. Ela oferece algumas ferramentas de uso gratuito em seu site, sendo umas delas o *SSL Checker*[6]. Nessa ferramenta, através do nome de domínio é realizada uma análise sobre os certificados utilizados pelo site. As primeiras informações mostradas são relacionadas ao certificado do servidor, possuindo algumas das informações já mostradas no visualizador de certificados do navegador, como o nome de domínio, a CA e a data de validade. Uma informação importante nesta análise é o tamanho da chave, nesse caso 2048 bits, considerado um tamanho suficiente de acordo com o NCSC-NL [NATIONAL CYBER SECURITY CENTRE 2019]. Em seguida, são mostradas informações sobre o nome, expiração, se o nome do certificado e o *hostname* são iguais, DNS e tipo do servidor e, por fim, sua cadeia de confiança. Sendo que é importante que esta possua CAs consideradas seguras, fazendo com que o certificado do servidor também seja considerado seguro.

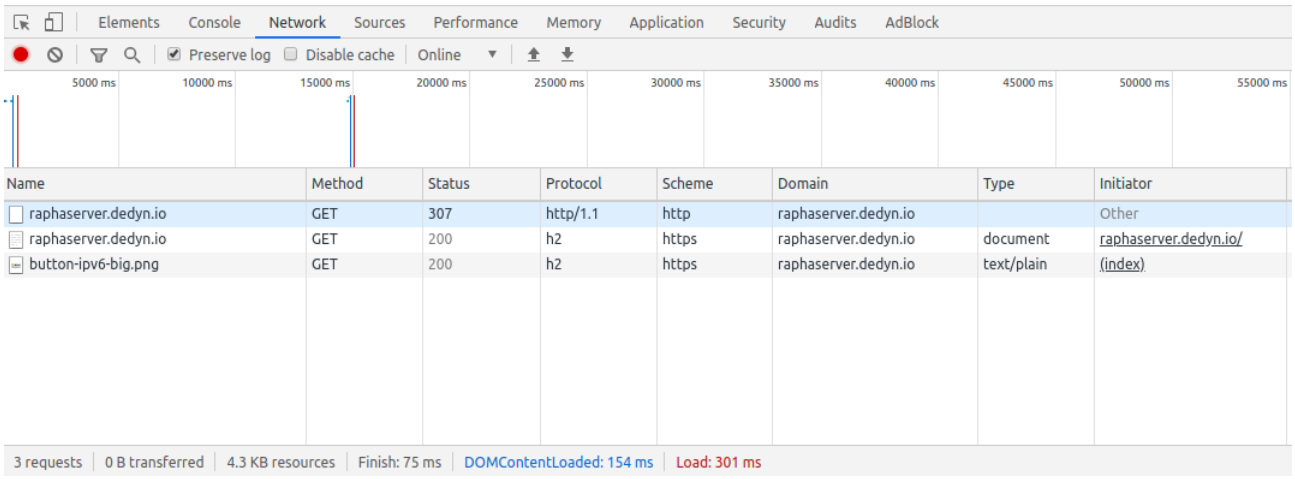
7.3. Conexão segura

Com a presença de um certificado válido e confiável, já é possível utilizar o HTTPS para acesso ao servidor. Através da ferramenta de desenvolvedor do navegador, certifica-se que a página possui um conexão HTTPS válida. Também é exibida a utilização de um certificado adequado e uma conexão com configuração correta, que utiliza uma versão atualizada do TLS e algoritmos para criptografia e autenticação seguros. É apresentada a utilização do P-384, um parâmetro para o algoritmo ECDHE considerado seguro para troca de chaves, segundo o NCSC-NL [NATIONAL CYBER SECURITY CENTRE 2019].

Por fim, pode-se comprovar que todos os recursos utilizados pelo site são acessados por meio de uma conexão segura, através do HTTPS. Isso está evidenciado na Figura 2, que mostra que a única imagem renderizada (button-ipv6-big.png) também foi requerida utilizando o HTTPS. Além disso, essa imagem mostra o fluxo de chamadas que ocorre ao tentar utilizar o site via HTTP. Ao tentar acessá-lo pelo HTTP ele retorna o status 307, que indica que o recurso da requisição foi temporariamente alterado, e então é redirecionado para uma requisição via HTTPS. Dessa forma, o site é acessível apenas via HTTPS, garantindo uma conexão mais segura.

[6]https://www.geocerts.com/ssl-checker

Figura 2. Redirecionamento de HTTP para HTTPS



Outro aspecto a ser observado é a utilização de compressão. O servidor implementado desabilita a compressão que, no caso do Nginx, é feita através do GZIP.

A fim de analisar outros aspectos de conexão segura, foi realizado um teste em uma das ferramentas disponibilizadas pelo *SSL Labs*[7]. O servidor implementado obteve uma nota total A+, sendo suas notas individuais todas maiores que 90.

Além das informações já apresentadas, este teste também exibe que o servidor impõe suas cifras e sua ordem de preferência. É possível ver que as cifras consideradas rápidas e seguras são preferidas, de acordo com o NCSC-NL [NATIONAL CYBER SECURITY CENTRE 2019].

Neste relatório, tem-se ainda uma visão de outras configurações. Dentre elas, é mostrado que o servidor não suporta compressão TLS, evitando que um invasor tenha acesso às informações sobre as partes secretas da comunicação criptografada. Além disso, é mostrado que o servidor suporta renegociação segura, presente na última versão do TLS (1.3). Entretanto, não permite renegociação iniciada pelo cliente, de forma segura ou insegura, evitando a possibilidade de ataques DoS. Foi verificado também que o servidor possui suporte ao OCSP *stapling* e que o 0-RTT está desabilitado. Esse teste indica também a utilização do HSTS, que possui um período de validade do cache (*max-age*) de pelo menos seis meses, que é o considerado suficientemente seguro pela NCSC-NL.

A versão TLS suportada é 1.3, sua versão mais recente, e é importante que o servidor desabilite as versões antigas. O teste realizado pela CDN77[8]mostrou que o servidor possui as versões 1.3 e 1.2 do TLS habilitadas e todas as outras versões, já descontinuadas, desabilitadas.

7.4. DNSSEC

Para realizar o teste do DNSSEC foi utilizada a ferramenta disponibilizada pelo DNSViz[9], que analisa a resolução do nome de domínio.

[7]https://www.ssllabs.com/ssltest/
[8]https://www.cdn77.com/tls-test
[9]https://dnsviz.net/

A autenticação começa pela zona ponto (.), que possui o *hash* da chave pública (DNSKEY) com uma borda dupla, que indica que foi designado como uma âncora de confiança. O registro *delegation signer* (DS) da zona pai é o DNSKEY da zona filha, servindo para assegurar autenticidade à delegação da zona, indicando qual chave pública pode ser utilizada. O processo segue passando por cada zona até chegar no nome de domínio final (raphaserver.dedyn.io), que será resolvido para o IPv4 (A) e IPv6 (AAAA). Nesse campo, é possível ver os dados relacionados à resolução do IPv4, no qual são mostrados o nome de domínio, o *time to live* (TTL), o tipo do registro sendo A para o IPv4, o valor do registro que contém o IPv4 a ser utilizado, os servidores autoritativos, seus nomes e seus identificadores, e, por fim, o *status* final *SECURE*, indicando que essa resolução é segura.

De forma semelhante ao IPv4, a resolução do IPv6 mostra as mesmas informações. Para o IPv6, o tipo de registro utilizado é o AAAA e seu valor é correspondente ao endereço IPv6. Como os servidores autoritativos utilizados são os mesmos, os dados se mantém, finalizado com o *status* final *SECURE*, indicando novamente que essa resolução é segura.

7.5. DANE

O mesmo site utilizado na geração do registro TLSA oferece uma ferramenta para testar o DANE[10]. Nesta ferramenta é necessário apenas colocar o nome de domínio e a porta utilizada na geração do registro. O teste é realizado primeiramente com IPv6 e a porta 443, sendo o resultado DANE OK, indicando que testou-se sua existência e validação. Entretanto, o segundo teste realizado, através do IPv4, retorna erro de *timeout*. Isso ocorre porque, como explicado na Seção 6.5, o servidor web foi instalado em uma rede doméstica e a provedora de Internet bloqueia essa porta, impossibilitando que o teste seja realizado.

7.6. Outras opções de segurança

Para analisar os protocolos restantes, foi utilizado o Mozilla Observatory[11], uma ferramenta de desenvolvimento *open source*. Nesse teste, o servidor obteve uma nota A+, passando em todos os 11 testes.

Ao analisar os testes realizados pelo Mozilla, verificou-se que foram explorados todos os cabeçalhos HTTP definidos pela OSE. O servidor utiliza os valores adequados para os cabeçalhos CSP, *X-Frame-Options*, *X-Content-Type-Options* e *Referrer-Policy*, recomendados pela OSE, além de alguns outros, como *X-XSS-Protection*, que pode oferecer proteção para usuários de navegadores mais antigos que ainda não suportam CSP.

7.7. Teste OSE

Após serem feitos todos os testes de forma separada, foi realizado o teste de todos os protocolos através do site *Internet.nl*. Apesar de terem sidos implementados todos os protocolos, a porcentagem mostrada pelo site é de 96%. Isso ocorre pelo bloqueio das portas 80 e 443 pela operadora, que faz com que os testes não consigam ser executados através do IPv4, uma vez que o site *Internet.nl* não permite portas diferentes das padrões de HTTP e HTTPS para executar os testes. Nesse caso, todos os testes são executados

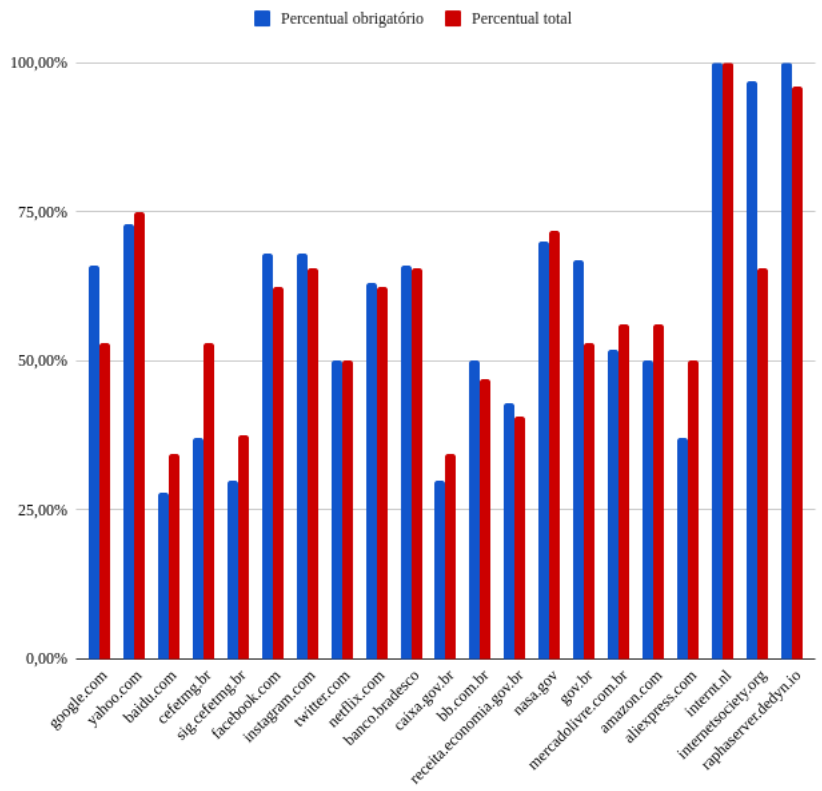
[10]https://www.huque.com/

[11]https://observatory.mozilla.org/

com IPv6, mas, ao realizar o teste de comparação do site do IPv6 com o IPv4, ele não obtém sucesso, uma vez o site não é alcançado via IPv4 nas portas padrões.

Todos os protocolos propostos pela OSE, com exceção da comparação do site do IPv6 com o IPv4, foram implementados corretamente. Comparando com os sites muito acessados já analisados nesse trabalho, pode-se ver na Figura 3 que o servidor desenvolvido nesse projeto possui uma taxa de implementação muito mais alta, ficando atrás apenas do *internet.nl*.

Figura 3. Comparação do servidor implementado



7.8. Outros padrões de segurança

Por fim, obteve-se outro relatório de segurança para o servidor web implementado por meio do site da ImmuniWeb[12], uma empresa global focada em segurança de aplicativos que oferece ferramentas gratuitas para diversos testes de segurança, no qual o servidor obteve a nota A.

Nos últimos 6 meses, apenas 46,6% dos servidores web alcançaram nota A nos testes realizados pela ImmuniWeb. Para servidores de e-mail, a porcentagem foi ainda menor, de apenas 10%, sendo que 48% obtiveram nota F. Dentre servidores de outras categorias, a taxa com nota A foi de 18,3%.

Além dos protocolos definidos pela OSE, existem outros padrões que visam garantir a segurança de um servidor. O relatório foi gerado por meio de testes de conformidade com base em diferentes padrões. O primeiro é o Padrão de Segurança

[12]https://www.immuniweb.com/

de Dados da Indústria de Cartões de Pagamento (PCI DSS), que foi desenvolvido para incentivar e aprimorar a segurança dos dados do titular do cartão e promover a ampla adoção de medidas de segurança de dados consistentes no mundo todo [PCI SECURITY STANDARDS COUNCIL 2018]. O servidor implementado atende a todos os itens definidos por este padrão, sendo que nos últimos 6 meses apenas 25,8% dos servidores testados estavam em conformidade.

Além desse padrão, foram realizados testes de conformidade com *Health Insurance Portability and Accountability Act* (HIPAA), *National Institute of Standards and Technology* (NIST) e melhores práticas da indústria.

8. Conclusão

Este trabalho teve como objetivo realizar um estudo de caso para analisar se alguns servidores web de sites conhecidos e muito acessados estão seguindo as normas e protocolos definidos pela OSE, e indicar melhorias a serem feitas para alcançar essas diretrizes. Além disso, propor um projeto de estrutura de um servidor web que implementa essas normas e protocolos.

Por meio da análise dos domínios apresentada nesse trabalho, foi possível obter o panorama geral de quais protocolos estão sendo implementados e como os domínios estão nesse âmbito. Com isso, foi possível ver como ainda há poucos sites que seguem essas boas práticas de segurança, deixando-os vulneráveis a diversos ataques. A partir disso, fica evidente a importância do projeto do servidor com os protocolos já implementados. A partir dos testes realizados, foi possível verificar como o servidor implementado obedece às normas e protocolos definidos, possibilitando a criação de servidores web mais seguros, utilizando recursos e serviços gratuitos disponíveis na Internet. Acredita-se que a reduzida adesão às diretrizes do projeto OSE seja mais por desconhecimento por parte dos administradores de servidores web do que por dificuldades na implementação das normas, portanto, as iniciativas de divulgação do projeto OSE feitas pela ISOC são fundamentais para melhorar a segurança na web.

Esse trabalho evidencia a existência de informação e recursos gratuitos para a implementação desses protocolos. A maior dificuldade encontrada durante a execução do projeto foi devida à utilização de um servidor doméstico, que possui as portas padrões bloqueadas, sendo que servidores corporativos não enfrentariam esse problema. A grande contribuição desse trabalho é mostrar a disponibilidade e facilidade da utilização dessas ferramentas.

A partir da importância do desenvolvimento desse projeto, um próximo passo envolveria implementar outros padrões de segurança, além dos definidos pela OSE. Além disso, expandir essa implementação para outros servidores web como o Apache, ainda muito utilizado, e também criar um *script* de automatização de configuração para esse servidor. Poderiam ser desenvolvidos também *scripts* de automatização de testes de segurança e testes de invasão.

Referências

Arends, R., Austein, R., Larson, M., Massey, D., and Rose, S. (2005). RFC 4033: DNS Security Introduction and Requirements.

BROADBAND COMMISSION (2019). *The State of Broadband Report 2019. International Telecommunication Union and United Nations Educational*.

Cloudfare (2020). *What is DNS cache poisoning? — DNS spoofing*. Disponível em: https://www.cloudflare.com/learning/dns/dns-cache-poisoning/. Acesso em: 04 jun 2020.

Cloudfare (2021). *What is a DNS MX record?* Disponível em: https://www.cloudflare.com/pt-br/learning/dns/dns-records/dns-mx-record/. Acesso em: 02 fev 2021.

Deering, S. and Hinden, R. (1998). RFC 2460: Internet Protocol, Version 6 (IPv6) Specification.

Dukhovni, V. and Hardaker, W. (2015). RFC 7671: The DNS-Based Authentication of Named Entities (DANE) Protocol: Updates and Operational Guidance.

Durumeric, Z., Kasten, J., Bailey, M., and Halderman, J. A. (2013). Analysis of the https certificate ecosystem. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, page 291–304, New York, NY, USA. Association for Computing Machinery.

Gont, F. (2014). *RFC 7217: A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)*.

IETF (2020). *Internet Engineering Task Force*. Disponível em: https://www.ietf.org/. Acesso em: 14 mar 2020.

ISOC (2020a). *Internet Society*. Disponível em: https://www.internetsociety.org/. Acesso em: 14 mar 2020.

ISOC (2020b). *Internet Society - Open Standards Everywhere Documentation*. Disponível em: https://github.com/internetsociety/ose-documentation/. Acesso em: 14 mar 2020.

Jankov, T. (2020). Nginx vs apache: Confronto entre servidores web. Disponível em: https://kinsta.com/pt/blog/nginx-vs-apache. Acesso em: 08 jun 2021.

NATIONAL CYBER SECURITY CENTRE (2019). It security guidelines for transport layer security (tls).

OWASP (2017). *OWASP Top Ten*. Disponível em: https://owasp.org/www-project-top-ten/. Acesso em 08 jun 2021.

PCI SECURITY STANDARDS COUNCIL (2018). *Requirements and Security Assessment Procedures*. Disponível em: https://www.pcisecuritystandards.org/document_library. Acesso em 11 fev 2021.

Rescorla, E. (2000). RFC 2818: HTTP Over TLS.

Rescorla, E. (2018). RFC 8446: The Transport Layer Security (TLS) Protocol Version 1.3.

Ribeiro, F. (2019). *Apenas 53% dos sites governamentais são seguros*. Disponível em: https://canaltech.com.br/espionagem/apenas-53-dos-sites-governamentais-sao-seguros-diz-relatorio-155859/. Acesso em: 07 fev 2021.