Análise de Certificados Digitais em Domínios Brasileiros

Matheus Aranha[1]
Diogo Pereira[1]
Artur Ziviani[1]
Fábio Borges[1]

[1]Laboratório Nacional de Computação Científica (LNCC)
25651-075, Petrópolis - RJ - Brasil

{maranhas,dpereira,ziviani,borges}@lncc.br

Abstract. *We introduce a security requirement and its security assessment for an Internet protocol. Specifically, this work presents a verification of the RSA keys of digital certificates present in the Brazilian domains that use the HTTPS protocol. Such verification depends on randomness in the generation of prime numbers. We use Graph Theory concepts to get three results based on the data we collected from hundreds of millions of domains. In the first result, we performed an iteration on the certificates, generating hundreds of millions of verifications. Luckily, we show that HTTPS is safe from this attack. In the second, we show that many domains share the same cryptographic key. In the third, we show that only 1% of the certification authorities are relevant.*

Resumo. *Apresentamos um requisito de segurança e sua avaliação de segurança para um protocolo da Internet. Especificamente, este trabalho apresenta uma verificação das chaves RSA dos certificados digitais presentes nos domínios brasileiros que usam o protocolo HTTPS. Tal verificação depende da aleatoriedade na geração de números primos. Utilizamos conceitos de Teoria dos Grafos para obtermos três resultados baseados nos dados que coletamos de centenas de milhões de domínios. No primeiro resultado, realizamos uma iteração sobre os certificados, gerando centenas de milhões de verificações. Felizmente, mostramos que o HTTPS está seguro a este ataque. No segundo, mostramos que muitos domínios partilham a mesma chave criptográfica. No terceiro, mostramos que apenas 1% das autoridades certificadoras são relevantes.*

1. Introdução

Fisicamente possuímos métodos que permitem a autenticidade de nossas ações e comprovação de nossa identidade, seja por um documento emitido por um órgão oficial, amplamente reconhecido, ou seja por uma assinatura física. De forma similar, os sites na internet também possuem formas que validam sua autenticidade, garantindo que a informação enviada pelo usuário ao site realmente será enviada ao destinatário correto. A forma mais utilizada para um site validar sua autenticidade é por meio da utilização de certificados digitais, certificados válidos garantem que o site em questão é realmente válido e seguro.

Certificados digitais são produzidos com algoritmos criptográficos assimétricos que geram as assinaturas digitais. Os algoritmos assimétricos mais usados na internet

são o RSA (Rivest, Shamir e Adleman) introduzido por [Rivest et al. 1978] e os baseados em curvas elípticas que foram introduzidas simultaneamente por [Koblitz 1987] e [Miller 1986] como uma alternativa eficiente para gerar algoritmos assimétricos.

Basicamente, os certificados digitais são documentos digitais que possuem características únicas, como a chave pública de um site, informações inerentes ao site para o qual o certificado foi emitido, seu período de validade e informações relacionadas a sua autoridade certificadora. Em resumo, pode-se adquirir um certificado digital dirigindo-se a uma autoridade de registro que coleta os dados para produção do certificado digital e verifica a validade de tais dados. A autoridade de registro transmite um arquivo com os dados para uma autoridade certificadora que assina tal arquivo com sua chave privada, gerando o certificado digital. Podemos verificar a validade do certificado porque as chaves públicas das autoridades certificadoras estão embutidas no software que usamos, por exemplo, o próprio sistema operacional ou algum navegador (*browser*).

Neste trabalho, analisamos somente certificados que utilizam o RSA como algoritmo de criptografia. Além de ser muito utilizado em assinatura de certificados digitais, o RSA também é utilizado na transmissão segura de dados em sistemas comerciais, privacidade e autenticidade de e-mails, sistemas de pagamentos, cartões de crédito, entre outros sistemas, por conta disso as vulnerabilidades presentes no RSA são muito estudadas [Boneh et al. 1999].

Um dos requisitos de segurança do RSA é a aleatoriedade na geração dos números primos, ou seja, entropia alta. Algoritmos de geração de números pseudoaleatórios de baixa qualidade acabam gerando números primos repetidos, provocando uma vulnerabilidade em protocolos baseados no RSA, que tem módulo $n = pq$ onde p e q são primos. Se pelo menos duas chaves públicas de RSA possuem um fator comum em seus módulos, o atacante que possuir tais chaves públicas pode utilizá-las para fatorar os módulos e consequentemente gerar a chave privada de cada um dos certificados. Usando esta técnica, [Lenstra et al. 2012] e [Barbulescu et al. 2016] conseguiram descobrir diversas chaves privadas geradas com RSA. No entanto, não fica claro se alguma das chaves comprometidas pertence ao domínio brasileiro, nem ao menos fica claro se alguma das chaves comprometidas era de certificados usados no HTTPS (*Hyper Text Transfer Protocol Secure*). Faz-se necessário uma avaliação de segurança nos certificados gerados com RSA para o protocolo HTTPS.

Inicialmente, levantamos a hipótese que as chaves comprometidas foram geradas para outros protocolos, usando algoritmos inapropriados como no fato das chaves fracas em ambiente Linux, relatado por [Yilek et al. 2009]. Felizmente, os domínios brasileiros com HTTPS passaram na avaliação de segurança. Note que tal avaliação de segurança deveria ser feita para cada novo certificado.

Outro ponto importante abordado neste trabalho está relacionado com a infraestrutura de chaves públicas, tais estruturas têm como principal núcleo as autoridades certificadoras, que são responsáveis pela manutenção dos certificados digitais [Braun et al. 2014]. Em particular, grande parte dos certificados digitais confiáveis atuais estão concentrados em uma pequena quantidade de autoridades certificadoras, não sendo necessário grande parte das autoridades certificadoras existentes, permitindo a emissão de certificados com um índice maior de confiabilidade [Braun and Rynkowski 2013].

Neste trabalho, coletamos e extraímos informações presentes nos certificados digitais dos domínios brasileiros (com extensão .br) com intuito de verificar este requisito de segurança das chaves RSA que são transmitidas nos certificados. Além da verificação das chaves, outro ponto foi a realização de uma análise destes certificados, permitindo ter uma visão geral de como os certificados estão distribuídos nos domínios. Também foi possível analisar a distribuição e importância das autoridades certificadoras utilizadas pelos domínios brasileiros.

As demais seções deste trabalho estão organizadas da forma descrita a seguir. Na Seção 2, apresentamos a metodologia utilizada para execução do trabalho, assim como uma breve descrição das técnicas utilizadas. Na Seção 3, apresentamos os resultados obtidos com a verificação das chaves, também são apresentadas análises sobre os dados coletados e análise dos grafos gerados com os dados e seus respectivos resultados. Por fim, na Seção 4, apresentamos a conclusão deste trabalho e possíveis trabalhos futuros.

2. Metodologia

Esta seção apresenta informações sobre a obtenção dos certificados, sobre o requisito de segurança e sua respectiva avaliação de segurança das chaves criptográficas do RSA no HTTPS.

2.1. Coleta dos Certificados Digitais e Extração das Informações

O processo de coleta dos dados dos certificados digitais que utilizam o RSA foi limitado somente para domínios que possuem extensão .br, registrados entre os anos de 2012 e 2013, disponíveis no site https://dnscensus2013.neocities.org.

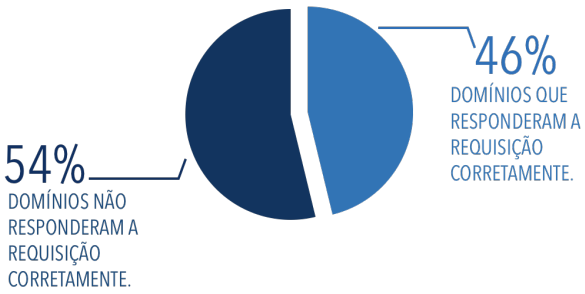

Figura 1. Taxa de sucesso e erro das requisições.

Para domínios brasileiros, foi totalizado uma quantidade de 572 506 domínios registrados na base de dados utilizada, onde apenas 264 511 responderam de forma satisfatória a requisição para obtenção das chaves públicas do RSA nos certificados, o restante dos domínios, totalizando 307 995 domínios, apresentaram erros de requisição (tempo expirado, domínio inválido, entre outros erros), impossibilitando a coleta dos dados dos certificados. A Figura 1 apresenta a relação entre sucesso e falhas na coleta das chaves pública do RSA nos certificados brasileiros.

Em relação aos domínios que responderam de forma satisfatória a requisição dos certificados digitais, realizamos uma categorização de acordo com os padrões utilizados pelo dominio.br, obtendo uma noção de como os domínios brasileiros que possuem certificados válidos estão distribuídos, tal categorização pode ser observada na Figura 2.

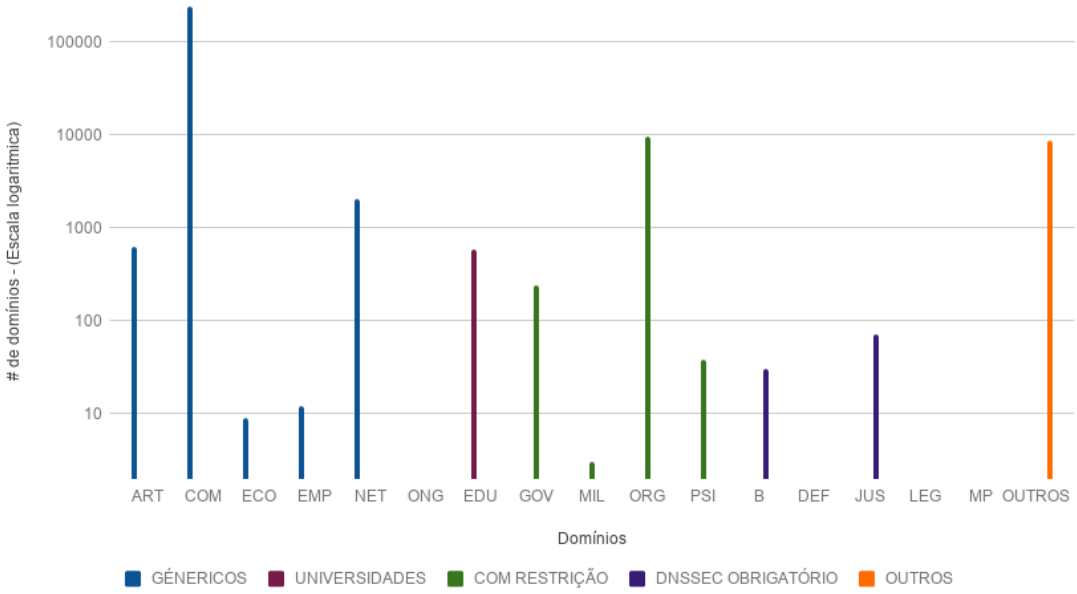

Figura 2. Categoria de domínios brasileiros em escala logarítmica dos certificados coletados.

Para o processo de categorização foram utilizados os padrões registrados pelo dominio.br, disponível em https://registro.br/dominio/categoria.html. Por isto, utilizamos as categorias "Genéricos", "Universidades", "Com restrição" e "DNSSEC obrigatório". Os domínios contabilizados na categoria "Outros" são domínios que não se enquadram nas categorias mencionadas.

Durante o processo de coleta dos certificados foram extraídas informações importantes para verificação e análise, tais como: autoridade certificadora, domínios associados ao certificado, validade do certificado e algoritmo de hash utilizado, além de expoente e módulo de cada chave pública do RSA. As informações extraídas foram armazenadas em um *dataset*, utilizado posteriormente no processo de verificação e análise dos dados.

Para a coleta, extração dos certificados e criação do *dataset*, escrevemos um *script* em Python que utiliza a biblioteca asn1crypto.x509 [Wbond 2018]. O *script* utilizado pode ser obtido em https://github.com/mattslv/rsa-sanity-check.

2.2. Avaliação de Segurança das Chaves RSA no HTTPS de Domínios Brasileiros

Após a extração do módulo de cada chave RSA nos certificados digitais, foi realizado a verificação das chaves RSA que consiste no cálculo do máximo divisor comum (MDC) de todos os módulos combinados dois a dois utilizando o algoritmo de Euclides, obtendo como saída os módulos cujo MDC fosse maior que 1. Lembre-se que cada módulo deveria ter no mínimo 1024 bits, ou seja, um número de aproximadamente 309 dígitos em decimal. No entanto, observamos que 101 certificados estão abaixo do número mínimo de bits aceitos para uma chave RSA. Tabela 1 mostra a frequência do tamanho dos módulos de chaves RSA nos certificados. [Borges de Oliveira 2017] e [Téllez and Borges 2018] apresentam uma relação entre o tamanho das chaves do RSA e seu respectivo nível de segurança.

Tabela 1. Tamanho dos módulos e suas quantidades nos certificados.

Tamanho dos módulos	Quantidade de certificados
512 bits	101 certificados
1024 bits	4 848 certificados
1040 bits	1 certificados
2018 bits	1 certificados
2046 bits	1 certificados
2048 bits	251 080 certificados
2058 bits	1 certificados
2096 bits	1 certificados
2432 bits	1 certificados
3072 bits	14 certificados
4096 bits	8 462 certificados

Apesar dos tamanhos de chaves inapropriados, observamos que todos os expoentes dos domínios coletados têm valores iguais a 65537, mostrando que os certificados digitais dos domínios brasileiros estão em conformidade com valores amplamente utilizado no RSA. [Lenstra et al. 2012] e [Barbulescu et al. 2016] encontraram expoentes com outros valores em outros protocolos. Felizmente, nenhum certificado usa a função de hash MD5. A Figura 3 mostra a porcentagem das frequências das funções de hash encontradas nos certificados.

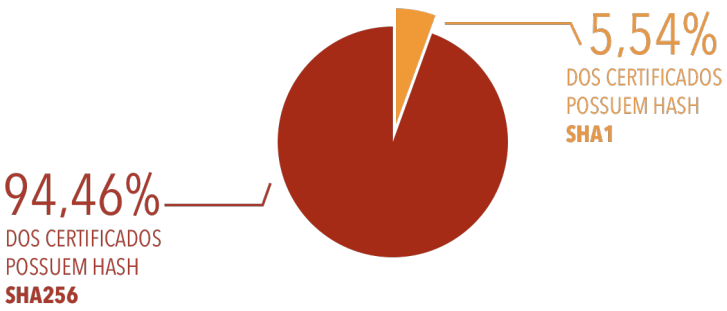

Figura 3. Algoritmos de hash utilizados nos certificados.

No total foram calculados 415 080 078 funções MDC. Note que o MDC tem complexidade polilogarítmica, sendo muito mais rápido que algoritmos de fatoração, e portanto, já foi usado em ataques ao RSA [Borges 2008]. De fato, este ataque apresenta uma forma de descobrir os fatores de módulos que eventualmente tenham algum fator em comum. Vários outros algoritmos criptográficos usam produtos de primos nos módulos [Borges et al. 2017], consequentemente, também podem ser atacados com a mesma estratégia. Neste caso, a segurança é baseada na aleatoriedade dos números primos que compõem os módulos.

Para o cálculo do MDC entre os módulos, escrevemos um *script* usando a biblioteca gmpy2 [Martelli 2017] que possibilita operações aritméticas de múltipla precisão, o *script* pode ser obtido em https://github.com/mattslv/rsa-sanity-check.

3. Discussões e Resultados

Esta seção apresenta discussões e resultados obtidos após a coleta e verificação dos dados extraídos dos certificados digitais. Foram realizadas análises das informações extraídas dos certificados digitais utilizando conceitos de Teoria dos Grafos e analisando a distribuição de grau, centralidade e modularidade dos grafos gerados com os dados.

Ao longo deste trabalho foram gerados grafos com representações distintas, utilizando parte das informações obtidas na Seção 2. Para cada grafo gerado, foram utilizadas diferentes representações de dados, possibilitando diferentes análises com o mesmo conjunto de dados. Especificamente, geramos três grafos a partir dos dados coletados.

No primeiro grafo, utilizamos apenas as chaves que possuem módulos únicos, com intuito de verificar a existência de certificados que possuem módulos com MDC maior que um. No segundo grafo, utilizamos os módulos e domínios dos certificados, com intuito de verificar se os domínios partilham ou não o mesmo módulo. No terceiro grafo, utilizamos como base os domínios e os nomes das suas autoridades certificadoras com a finalidade de verificar a quantidade de domínios por autoridade certificadora.

Tais representações através dos grafos podem ser observadas com mais detalhes nas seções abaixo, bem como seus respectivos resultados.

3.1. Primeira Representação por Grafo

Na primeira representação utilizada na análise, consideramos como nós os módulos únicos extraídos dos certificados digitais. Consideramos como arestas a existência de um MDC maior que 1 entre ambos.

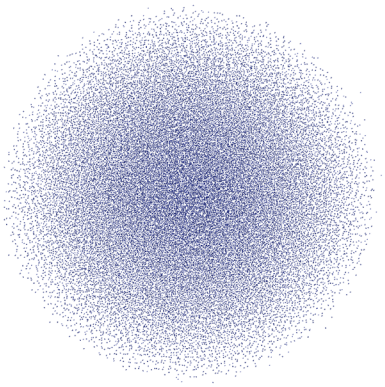

Figura 4. Grafo gerado pelo MDC entre os módulos únicos.

Durante o processo de verificação do requisito de segurança das chaves não foi encontrado MDC maior que um entre os módulos obtidos, desta forma, foi obtido um grafo com um total de 28 813 nós e nenhuma aresta entre os nós, com grau médio igual a zero, i.e., sem conexão. A Figura 4 apresenta o grafo obtido. É notável observar na figura que foi obtida uma nuvem desconexa de pontos, não existindo componentes ligados entre os nós do grafo.

Para o âmbito de segurança, temos um ótimo resultado. Mostrando que os algoritmos utilizados para geração de números aleatório nos domínios brasileiros são satisfatórios, e consequentemente, os domínios não são vulneráveis entre si a esta classe de

ataques. No entanto, quanto maior o número de módulos coletados, o ataque efetua mais verificações aumentando a chance de sucesso do ataque. Portanto, estamos trabalhando para coletarmos todos os módulos presentes no HTTPS.

3.2. Segunda Representação por Grafo

Foi gerada um grafo bipartido, onde os nós utilizados na primeira representação foram trocados de módulos únicos por módulos e endereços dos domínios (URLs - *Uniform Resource Locators*) extraídos dos certificados digitais. Com essa nova representação, foi obtido um grafo com um total de 292 824 nós e 264 511 arestas, com grau médio igual a 0,903. Lembre-se que um grafo completo tem grau 1.

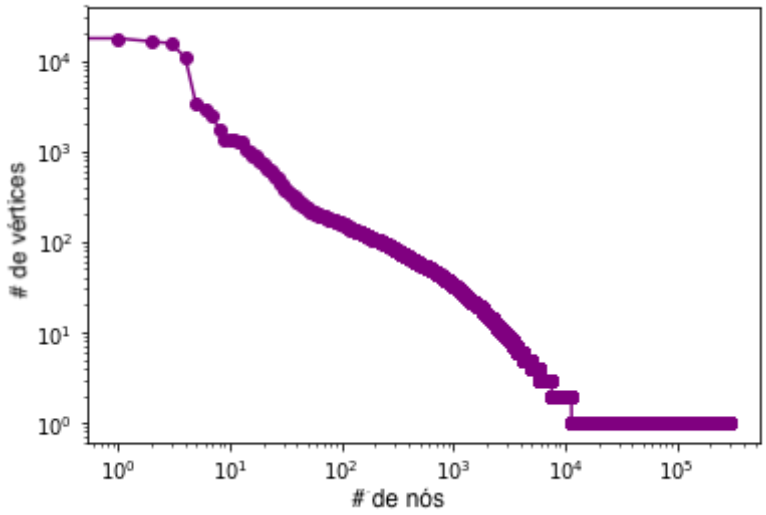

Figura 5. Distribuição de grau do grafo gerado em escala logarítmica.

Por meio da Figura 5, podemos observar a distribuição de grau gerada. Analisando a imagem, podemos perceber muitos nós possuindo uma concentração muito alta de arestas, ou seja, existem muitos módulos que possuem conexão com vários domínios, mostrando que uma grande quantidade de domínios brasileiros (na grandeza de 10^4) partilham o mesmo módulo. Consequentemente, quem tem conhecimento uma única chave destas quatro maiores concentrações, tem acesso as transações de dezenas de milhares de sites brasileiros que usam o HTTPS.

[Barabási and Pósfai 2016] apresentam que este tipo de grafo segue um modelo de rede complexa de escala livre regida por uma lei de potência, sendo bastante vulnerável a ataques direcionados. Sendo assim, um ataque direcionado aos nós que possuem uma grande quantidade de domínios associados podem fazer um grande estrago, por exemplo, o vazamento da chave privada ou a fatoração de um módulo da chave do RSA associado a milhares de domínios causaria um grande impacto para estes domínios.

Apesar de a grande maioria dos domínios terem apenas um módulo, para a área de segurança, o ideal seria a existência de um módulo diferente para cada domínio registrado, não havendo este compartilhamento de módulos entre os domínios.

3.3. Terceira Representação por Grafo

Utilizamos os nomes das autoridades certificadoras e os domínios como nós para gerar um grafo bipartido. As arestas são relações entre autoridades e módulos. A partir desta representação, obtivemos um grafo com 29 161 nós e 28 321 arestas, com grau médio igual a 0,971, i.e., próximo do grau de um grafo completo.

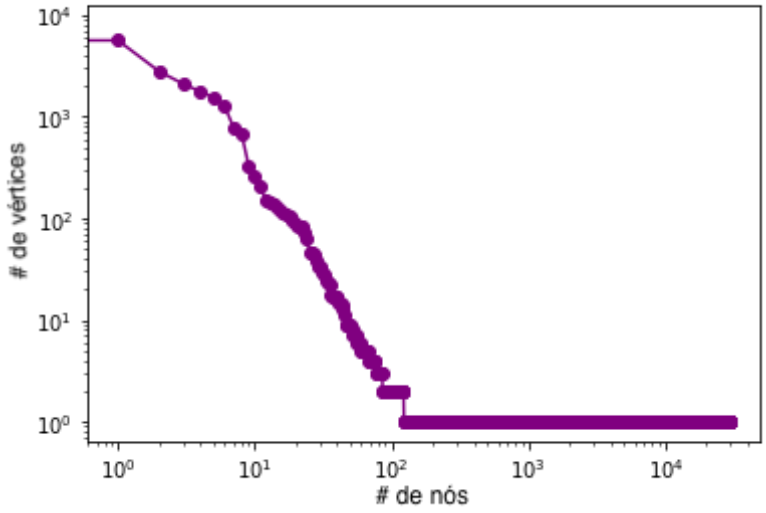

Figura 6. Distribuição de grau do grafo gerada em escala logarítmica.

É possível observar na Figura 6 que existem uma pequena quantidade de autoridades certificadoras que são responsáveis pela certificação da grande maioria dos domínios brasileiros. Também é possível visualizar que o grafo segue um padrão parecido com o do grafo anterior, ou seja, o grafo também é suscetível a ataques direcionados, sendo prejudicial se um atacante direcionar um ataque para uma autoridade certificadora responsável por muitos certificados digitais. Contrariamente, é muito melhor que se tenha poucas autoridades para confiar. [Braun and Rynkowski 2013] defendem que é muito mais fácil auditar e confiar em um número pequeno de certificadoras do que em um número muito alto.

Outro ponto importante a ser observado, caso 99% das autoridades certificadoras menos influentes que atuam nos domínios brasileiros fossem retiradas, aproximadamente 90% dos domínios que possuem certificados digitais ainda seriam certificados por uma autoridade válida, corroborando o ponto mostrado por [Braun and Rynkowski 2013]. Em particular, os certificados poderiam ser distribuídos baseados na localidade geográfica, seguindo um modelo similar ao DNS (*Domain Name System*).

Outra análise realizada para este grafo foi o cálculo de modularidade, onde foi obtido uma modularidade igual a 0,858 com um total de 849 comunidades formadas, mostrando que o grafo obtido tem alta chance de formar comunidades com nós de características parecidas, ou seja, grandes autoridades certificadoras tendem a continuar atuando e certificando grande parte dos certificados digitais. A distribuição de nós e modularidade pode ser observada na Figura 7.

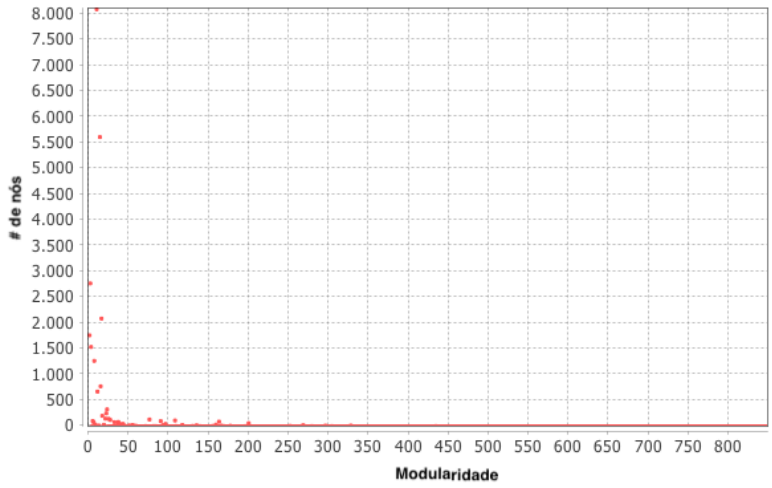

Figura 7. Distribuição de nós por modularidade do grafo gerado.

4. Conclusão e trabalhos futuros

Este trabalho apresenta a realização da avaliação de segurança de um requisito de segurança das chaves do RSA presentes nos certificados digitais dos domínios com extensão .br. No processo de verificação, fizemos uma análise através de Teoria dos Grafos e encontramos um resultado diferente de outros trabalhos na literatura. A diferença deve ser devida aos algoritmos de geração de números pseudoaleatórios de outros protocolos.

Mostramos na seção 3.1 que os domínios brasileiros estão livres entre si desta classe de ataques utilizando o módulo das chaves nos certificados. Porém, é preciso realizar esta verificação em um escopo maior, pois a amostra utilizada é relativamente pequena para tirar conclusões definitivas.

Mostramos na seção 3.2 que grande parte dos domínios brasileiros partilham os mesmos módulos e consequentemente os mesmos certificados, o que gera um grande problema de segurança, bastando que a chave privada de apenas um seja exposta para prejudicar os demais domínios pertencentes ao mesmo grupo.

Por fim na seção 3.3, mostramos a existência de uma concentração muito grande das autoridades certificadoras, sendo possível visualizar que grande parte dos domínios brasileiros são certificados por poucas autoridades. Temos que 99% das autoridades certificadoras dos certificados coletados são irrelevantes atualmente para manter os domínios brasileiros certificados.

É interessante realizar as mesmas análises para todos os domínios da Internet, principalmente a verificação das chaves RSA. Já estamos trabalhando nesta direção. Estamos buscando também a realização de outras representações, através de grafos, com os dados extraídos.

Agradecimentos

Gostaríamos de agradecer ao Pronametro, ao CNPq, e à FAPERJ pelo apoio no desenvolvimento deste trabalho.

Referências

Barabási, A.-L. and Pósfai, M. (2016). *Network science*. Cambridge University Press, Cambridge.

Barbulescu, M., Stratulat, A., Traista-Popescu, V., and Simion, E. (2016). Rsa weak public keys available on the internet. In *International Conference for Information Technology and Communications*, pages 92–102. Springer.

Boneh, D. et al. (1999). Twenty years of attacks on the rsa cryptosystem. *Notices of the AMS*, 46(2):203–213.

Borges, F. (2008). Um novo algoritmo probabilístico para fatoração de inteiros com primos relativamente distantes. In *Anais do VIII SBSeg - Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, pages 269–270, Gramado - RS. Sociedade Brasileira de Computação (SBC).

Borges, F., Lara, P., and Portugal, R. (2017). Parallel algorithms for modular multiexponentiation. *Applied Mathematics and Computation*, 292:406–416.

Borges de Oliveira, F. (2017). *Selected Privacy-Preserving Protocols*, pages 61–100. Springer International Publishing, Cham.

Braun, J. and Rynkowski, G. (2013). The potential of an individualized set of trusted cas: Defending against ca failures in the web pki. In *Social Computing (SocialCom), 2013 International Conference on*, pages 600–605. IEEE.

Braun, J., Volk, F., Classen, J., Buchmann, J., and Mühlhäuser, M. (2014). Ca trust management for the web pki. *Journal of Computer Security*, 22(6):913–959.

Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209.

Lenstra, A., Hughes, J. P., Augier, M., Bos, J. W., Kleinjung, T., and Wachter, C. (2012). Ron was wrong, whit is right. Technical report, IACR.

Martelli, A. (2017). gmpy2 library. https://github.com/aleaxit/gmpy.

Miller, V. S. (1986). Use of elliptic curves in cryptography. In *LNCS 218 on Advances in Cryptology—CRYPTO 85*, pages 417–426, New York, NY, USA. Springer-Verlag New York, Inc.

Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126.

Téllez, C. and Borges, F. (2018). Trade-off between performance and security for supersingular isogeny-based cryptosystems. *Anais do Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSeg)*, pages 113–126.

Wbond (2018). asn1crypto library. https://github.com/wbond/asn1crypto.

Yilek, S., Rescorla, E., Shacham, H., Enright, B., and Savage, S. (2009). When private keys are public: Results from the 2008 debian openssl vulnerability. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, IMC '09, pages 15–27, New York, NY, USA. ACM.