

**ATAQUES A CONTAS DE SERVIÇOS ONLINE**

SÃO CAETANO DO SUL/SP

2019

**ANTONIO DUARTE DIAS NETO**

**IGOR VINÍCIUS MOREIRA**

**MARCOS GONÇALVES SIQUEIRA**

**MIKAEL AKIRA SHISHITO MATOS**

**VITOR KIOMASSA KINA**

**YAN MOURA VIRGINIO**

**ATAQUES A CONTAS DE SERVIÇOS ONLINE**

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia de São Caetano do Sul, sob a orientação do Professor Dr. Jacinto Carlos Ascêncio Cansado, como requisito parcial para a obtenção do diploma de Graduação no curso de Segurança da Informação.

SÃO CAETANO DO SUL/SP

2019

**SUMÁRIO**

[INTRODUÇÃO 3](#_Toc24048847)

[Justificativa 4](#_Toc24048848)

[Objetivo 5](#_Toc24048849)

[Metodologia 6](#_Toc24048850)

[1. Contas e autenticação 7](#_Toc24048851)

[2. Explorando vulnerabilidades em contas 9](#_Toc24048852)

[2.1 Banco de Dados 9](#_Toc24048853)

[2.2 Hashcat 9](#_Toc24048854)

[2.3 SentryMBA 9](#_Toc24048855)

[3. Soluções de segurança 12](#_Toc24048856)

[3.1 Método “Anti-Automação” 12](#_Toc24048857)

[CONSIDERAÇÕES FINAIS 15](#_Toc24048858)

[REFERÊNCIAS 16](#_Toc24048859)

INTRODUÇÃO

As pessoas estão cada vez mais dependentes da tecnologia, seja para o trabalho ou apenas para uso pessoal, e muitos dos sites acessados necessitam criar uma conta de usuário, seja para uma rede social, um *streaming* de filmes ou um de música, tornando essencial ter senhas que sejam fortes e seguras para que dificultem o acesso de alguém não autorizado. Portanto é necessário saber como é possível prevenir ou diminuir as possibilidades de comprometimento da conta, para isso é preciso tomar medidas de segurança essenciais, como usar senhas diferentes em sites diferentes, usar senhas longas e com caracteres especiais.

As aplicações *web* são o principal alvo quando é falado sobre autenticação de contas utilizando e-mail, usuário ou senhas, isso ocorre devido ao alto nível de acesso a informações pessoais quando existe o sucesso ao ingresso a uma conta pessoal, podendo render lucro para o invasor de diversas formas e prejudicar o proprietário da conta acessada, tanto em termos de privacidade quanto financeiros. De acordo com o ramo do site alvo, os invasores podem conseguir dados de cartões de crédito, informações pessoais e outros dados sensíveis que podem compor um cenário de ataque e causar impactos significantes. As páginas *web* podem estar vulneráveis por muitas razões, em função, por exemplo, do desenvolvedor da página não usar comandos e precauções necessárias para manter o site confiável e seguro. Segundo o Open Web Application Security Project (OWASP, 2019), em 2017 os ataques mais críticos em aplicações *web* foram a injeção de Structured Query Language (SQL), que tem como objeto aproveitar de falhas de bancos de dados através a injeção de comandos SQL, o segundo ataque mais crítico é a quebra de autenticação, onde a autenticação e gerenciamento de sessão são implementadas incorretamente.

É importante para as empresas ter formas de mitigar os impactos de um ataque a uma aplicação *web*, segundo Gênesis Rivas (2018), as etapas importantes para a proteção de aplicações *web* são o uso de criptografia e configurar o servidor *web* para redirecionar para uma página *web* com criptografia; desenvolver um sistema de redefinição de senha segura; realizar uma avaliação de risco e desenvolver ou utilizar uma ferramenta de segurança de aplicativos da *web*.

Justificativa

O estudo do comprometimento de contas é importante pois muitos usam o mesmo *e-mail* e palavra-chave (password) em diversos sites, se sujeitando à possibilidade de terem seus dados privados descobertos por indivíduos “mal intencionados” que tenham acesso a um único site. Isso pode ser analisado segundo Glazier W. e Dhiman em Automation Attacks at Scale - Credential Exploitation (2017), onde na grande maioria dos dados que foram analisados, tinham sido usados em grandes sites famosos que tiveram seus dados vazados.

É importante ressaltar que esse tipo de ataque pode se tornar grave uma vez que pode atingir milhares de pessoas, tudo depende da segurança do site alvo e dos dados de usuário que as pessoas usam no site em questão. Por isso é recomendado por diversos especialistas que a sua senha não seja usada em mais de um site e que ela seja uma senha complexa, porém a complexidade da senha tem estado em discussão em detrimento do tamanho ou do uso de variações de caracteres.

Hoje em dia muitos sites ainda colocam a autenticação baseada em *e-mail* e senhas, por esse fator e através dos programas corretos, como o SentryMBA é possível acontecer o que foi demonstrado por Glazier W. e Dhiman (2017), em que houve a manipulação de dados de outros sites para acesso de contas ao site alvo, também foi verificado que mesmo os acessos legítimos constavam em outros sites, mas a quantidade desses dados em outros sites era bem menor em relação aos acessos tentados pelo programa em questão.

Com um ponto de vista da área da segurança, é sempre necessário estar atualizado aos meios de proteção atuais e buscar uma evolução constante, pois os ataques sempre estão evoluindo. De acordo com Glazier W. e Dhiman em Automation Attacks at Scale - Credential Exploitation (2017), uma tática de defesa ideal é aquela em que não alerte a presença da defesa para os atacantes e que não seja apenas um bloqueio, pois apenas isso pode fazer com que os atacantes criem técnicas para ultrapassar a segurança.

Para Feldmeier, D. e Kam (1990), as senhas são talvez os métodos mais utilizados de autenticação do usuário. Embora haja mais segurança em mecanismos de autenticação, por exemplo: *token de hardware*, autenticação baseada em impressão digital ou certificação, a autenticação baseada em senha continua a ser o mecanismo amplamente utilizado devido ao seu baixo custo e conveniência. Como senhas devem ser memorizadas, elas são geralmente escolhidas de um pequeno domínio, que permite a montagem de um ataque de dicionário em sistemas baseados em senha e outras formas de ataque. Baseado nesta informação, este trabalho de graduação, propõe a seguinte reflexão: é possível demonstrar proteção total do sistema de autenticação do usuário de acordo com o método de ataque tratado no trabalho?

Objetivo

O trabalho tem por objetivo explicar passo-a-passo sobre como as contas de serviços *online* podem ser invadidas, através de um simples software, e identificar as formas de prevenção, tanto aquelas que seriam realizadas pelas empresas quanto pelo usuário comum (pessoa física). Outro objetivo é o de demonstrar a importância de as partes entenderem e conscientizarem sobre esse assunto, pois um problema como esse pode causar um grande impacto para a organização e para o consumidor daquele serviço.

Este trabalho aborda mais profundamente o processo em que existe o acesso aos dados das contas de usuários e porque é imprescindível existir a conscientização sobre esse tema, tanto para informar o usuário do serviço quanto para que as empresas consigam desenvolver novas técnicas para defender seus dados e barrar acessos não autorizados.A intenção seria contribuir com desenvolvedores de forma geral, de modo que os conhecimentos que estarão disponíveis neste trabalho de graduação ajudem no desenvolvimento de sites, ou outros tipos de plataformas de serviços, proporcionando aos desenvolvedores formas alternativas para aplicação útil e mais rápidas na identificação e solução para as invasões ocorridas; e buscar esses resultados através de uma descrição detalhada das diversas maneiras de se obter acesso às contas privadas de usuários nos diversos tipos de serviços disponibilizados via internet, bem como a mitigação recomendada.

Metodologia

Neste trabalho de graduação, será explorado o tema proposto através de um cenário prático para demonstração do método de invasão, e explicado como prevenir tal método falado anteriormente, junto com o cenário, utilizando como base pesquisas bibliográficas em artigos, livros, sites acadêmicos e corporativos.

Este documento é composto de quatro capítulos, sendo o primeiro utilizado para a compreensão do funcionamento da autenticação em contas, o segundo irá focar nas ferramentas utilizadas, o terceiro será em um cenário prático do projeto e o último tratará das formas de mitigação.

1. Contas e autenticação

As contas na internet podem ser classificadas como uma forma de identificação, para que assim o sistema saiba quem você realmente é e consiga salvar seus dados e informações, além de protegê-los. Para poder analisar essa ação de uma forma mais clara, será utilizada a analogia de uma blitz policial, quando existe a solicitação de seus documentos por parte do policial, é como se o site, programa ou aplicativo em questão estivesse requisitando seus dados para verificar que você é quem diz ser, após o fornecimento desses dados, uma checagem interna é feita em ambos os casos. No caso real, é verificado no sistema governamental se o documento é válido, enquanto no caso de contas virtuais essa verificação pode ser feita de algumas formas diferentes, mas todas com o mesmo princípio: checar se os dados que você forneceu conferem com os dados necessários para acessar ou validar informações. De acordo com Renaud (2004, p. 2) “Todos os sistemas de segurança vão perguntar para a pessoa se identificar“.

Nessa verificação de identidade virtual, o modo mais comum é o que utiliza um e-mail ou usuário e também uma senha, isso significa que para ter acesso aos dados, é necessário possuir no mínimo duas informações, onde a senha é sempre extremamente pessoal e que só o proprietário daquela conta deveria possuir a informação, porém, em muitos casos como Felten e Gaw (2006, p.1) relata: “Quando as pessoas utilizam as mesmas senhas em várias contas, elas aumentam sua vulnerabilidade; o comprometimento de uma senha pode ajudar um atacante a assumir várias contas.”, isso faz com que a segurança de suas contas esteja em risco, mais adiante isso será abordado profundamente neste trabalho. Segundo Cristofaro et al (2013, p. 1), 20% das senhas são cobertas em uma lista com apenas 5,000 senhas. Por conta desse hábito dos usuários, passaram a existir alguns fatores extras para aumentar a segurança das contas, dentre eles, podemos citar a autenticação de dois fatores.

A autenticação de dois fatores consiste em uma segunda etapa além do padrão exigido de e-mail ou usuário e senha, essa etapa também consiste em validar o acesso à conta através de algo que somente o proprietário deveria ter acesso e pode ocorrer de modos distintos e pode ser realizada de duas formas diferentes, segundo Osório e Heinen (2004, p.2), conforme citado por Abbas (1994). Online e offline conforme explicado abaixo:

Na autenticação off-line, a assinatura é feita pelo usuário em uma folha de papel, que é posteriormente digitalizada e enviada para o sistema que realiza a autenticação. Na autenticação online, a assinatura é feita diretamente sobre um dispositivo de hardware, como uma mesa digitalizadora ou um tablet. Além destes dispositivos, podem ser utilizados para a autenticação on-line de assinaturas computadores do tipo Handheld, que permitem a escrita diretamente sobre uma tela sensível. O reconhecimento on-line de assinaturas permite que sejam utilizadas diversas informações temporais e dinâmicas relativas à assinatura, como a velocidade da caneta e a trajetória, que permitem que se obtenham melhores resultados no processo de autenticação.

Os dois modos mais populares e que podem ser verificados em diversos sites nos dias de hoje, são validações via um código que é recebido pelo proprietário. Ele recebe esse código, geralmente composto por números ou letras, em algum dispositivo confiável, podendo ser através de um SMS em seu número de telefone cadastrado no site ou em programa de terceiros que gere esse código, como o Google *Authenticator* faz. Como Cristofaro et al (2013, p. 1) menciona, “Autenticação em dois passos foca em melhorar a resiliência da autenticação baseada em senha solicitando que os usuários providenciem um passo adicional de verificação, por exemplo, um código gerado por um token de segurança”.

2. Explorando vulnerabilidades em contas

Entendendo como funciona autenticação nas contas de serviços online, como explicado no capítulo anterior, no decorrer deste capítulo demostra-se como essa autenticação pode ser violada. Pensando que uma conta é autenticada através de usuário e senha, a proposta é usar um banco de dados para que essa senha seja descoberta.

2.1 Banco de Dados

Um banco de dados pode ser um arquivo com milhares de informações armazenadas, segundo Korth “um banco de dados é uma coleção de dados inter-relacionados, representando informações sobre um domínio especifico”, ou seja, sempre que for possível agrupar informações que se relacionam e tratam do mesmo assunto, pode-se dizer que há um banco de dados. No caso deste trabalho será utilizado um arquivo com milhares de possibilidades de usuário e senha, um agente mal intencionado pode utilizar facilmente de um banco de dados roubado/vazado para conseguir acesso a uma conta online, tendo em vista que isso é algo ilegal, este trabalho utiliza um banco de dados criado para demonstração de como esse ataque ocorreria em um cenário real.

2.2 Hashcat

Em banco de dados, há a possibilidade de a senha dos usuários estarem criptografadas e portanto é necessário a utilização de um software para a quebra da criptografia, aqui será utilizado o Hashcat, que tem como função realizar a quebra de senhas criptografadas no banco de dados.

O Hashcat pode quebrar algoritmos de criptografia como por exemplo MD5, SHA-256, SHA-512, DES, SHA1 e sua velocidade de quebra dependerá da velocidade de processamento da CPU ou GPU.

2.3 SentryMBA

Atualmente os ataques automatizados ocorrem por meio de programas feitos exclusivamente para testar diversos e-mails e senhas no site alvo, dessa forma os atacantes podem apenas deixar o programa ligado em um VPS, com isso o programa irá ficar ligado 24h por dia, 7 dias por semana.

Dentre esses programas, o foco se encontrará no SentryMBA, por ser um dos principais e mais famosos softwares utilizados para esse tipo de ataque.

O SentryMBA é um programa gratuito desenvolvido para Windows e que por ser extremamente configurável, o torna utilizável para os mais diversos tipos de sites e através desses dois fatores é muito utilizado mesmo após anos de sua criação.

Para abordá-lo, será feita uma análise de seu funcionamento para que toda empresa possa analisar seus métodos de autenticação.

Para começar, vale ressaltar que a base de utilização do programa está em três pilares:

● Combolist ou wordlist: é uma lista que possui (geralmente em casas de milhares ou milhões) nomes de usuário (ou e-mails) e senhas que geralmente foram geradas a partir de bancos de dados que vazaram de sites. Essas listas possuem a senha em texto puro.

● Config: é o arquivo de configuração do SentryMBA para um determinado site alvo, ele é gerado e salvo pelo próprio programa em um arquivo .ini, para que esse arquivo seja criado, é necessário analisar os métodos de autenticação no site alvo através de um sniffer juntamente com o SentryMBA para que os parâmetros necessários sejam realizados.

● Proxy: é uma lista de IPs que faz o software ter um IP diferente para cada tentativa de autenticação, a intenção disso é que o sistema de segurança não perceba que o site está sob ataque e assim evitar algum tipo de bloqueio ao IP verdadeiro.

Como demonstração de criação de uma Config para o SentryMBA, será utilizado o site Tidal, a primeira etapa afim de obter êxito no ataque é a criação de uma conta comum no site escolhido. Com a utilização do sniffer, ou rastreador de pacotes será feita a autenticação da conta normalmente com o intuito de entender como o site realiza a checagem de que os dados estão corretos ou não, essa é uma etapa de extrema importância para a configuração, dentro do programa delimita-se qual é a palavra-chave que irá identificar o acesso concedido a uma conta e qual irá identificar que os dados estão incorretos.

É nessa identificação que o software separa as contas em que obteve êxito, dessa forma o real atacante pode fazer uso delas.

Ainda na etapa de configuração o atacante pode não querer apenas saber se os dados estão corretos, mas sim buscar mais informações da conta, como se possui algum tipo de saldo, algum cartão de crédito vinculado ou qualquer coisa que seja valiosa para ele e isso também é realizado da mesma forma que mostrada anteriormente, porém nessa etapa não apenas é feito o SentryMBA entender que ele está no local correto do site, como também é possível dar instruções para que ele capture os dados que estão nos campos que delimitamos para ele capturar.

Esse banco de dados poderá ser usado em um software, onde serão efetuados vários pedidos de autenticação de conta ao site com milhares de possibilidades de usuários e senhas do banco de dados até que o acesso ao site seja autenticado com a possibilidade certa.

O SentryMBA funciona com uma pré-configuração, ou seja, é necessário configurá-lo para atacar um site específico, pois cada site pode possuir uma característica própria no momento da autenticação. Para saber exatamente como o site gera essa autenticação, é necessário utilizar um sniffer.

Depois de configurado, o SentryMBA necessita de uma wordlist para tentar obter acesso as contas. Neste trabalho a wordlist será gerada manualmente, porém é possível obter wordlists facilmente na internet.

Um rastreador de pacotes monitora o tráfego da rede, conseguindo capturar qualquer informação que esteja passando por ela. Assim, é possível capturar as informações de como é feita a autenticação no nosso alvo. Será utilizado o Fiddler, que permitirá a exibição do conteúdo de uma sessão da Web como URLs, protocolos, redirecionamentos, e especificamente a autenticação de login.

2.3.1 Configurações no SentryMBA

Será tocado brevemente sobre as configurações do programa SentryMBA, para deixar um melhor entendimento sobre o mesmo, e sobre o processo como um todo.

Ao abrir o programa, uma de suas partes mostra as opções de configuração, com cinco abas. Elas são:

* + - Na aba superior:
      * Start: Começa o processo de ataque.
      * Abort: Encerra o processo de ataque.
      * Site: Onde é explicitado o URL do site alvo.
      * Switch Site: Pode ser especificado aqui outro site para ser atacado.
      * Progress: Irá mostrar o progresso atual.
      * List: Informa qual lista estará sendo utilizada.
    - Aba General (Geral):

Nesta tela de configurações gerais, as seguintes configurações podem ser alteradas para alcançar o resultado esperado de forma mais eficiente.

* + - * Site Settings:
        + Timeout(s): Essa caixa de texto permite especificar o tempo que o SentryMBA utilizará, em segundos, para tentar realizar a conexão com o servidor. Valores maiores significam mais tempo em cada conexão, enquanto valores menores irão fazer com que o ataque seja mais rápido, porém isso implicará na possibilidade de mais erros de limite de tempo atingido quando havia sim a possibilidade de uma conexão bem-sucedida.
        + Bot relaunch delay(s): Especifica, também em segundos, o tempo que a aplicação levará para enviar um novo bot ao servidor, após o retorno do último. É útil para APIs que limitam a quantidade de solicitações por segundo. É fácil de detectar se uma API possui esse tipo de característica, pois após uma tentativa de ataque, serão mostrados quantos bots foram perdidos.
        + Resolve Hostname: Irá resolver o nome do host, ou hostname. Isto significa que o hostname será transformado em seu endereço IP equivalente. Pode ser útil dependendo do site, com uma chance de acelerar o ataque.
      * Combo Settings:
        + <USER>:<PASS> filter: Uma ferramenta poderosa. Pode filtrar os usuários e senhas com diversos argumentos. É possível desabilitar um caractere que é de conhecimento do atacante que é recusado pelo site, definir o comprimento mínimo e máximo da string do usuário, parametrizar somente o uso de passes com letra maiúscula ou minúscula, entre outros.
        + <EMAIL> filter: Com esta opção estando ativada, o SentryMBA só irá utilizar da lista de passes, aquelas onde o campo da esquerda, ou seja o campo de usuário (só lembrando, os passes são formatados da seguinte forma: “usuário:senha”) for um e-mail válido. Geralmente, na maioria das combolists, o campo da esquerda é preenchido inteiramente de e-mails, então esta opção pode ser um pouco deixada de lado.
      * General Settings:
        + Save automatically valid usernames and expired combos: Irá salvar automaticamente nomes de usuário válidos e combinações expiradas.
        + Annoying sound on Hit: Toca um som previamente especificado toda vez que um passe é bem-sucedido. Este som deve estar em formato .wav.
        + Popup Memo containing Hit debug information: Assim que uma combinação funcionar, informações serão mostradas sobre o evento, que podem auxiliar a configurar melhor um futuro ataque.
        + Minimize to Tray: Minimiza o programa para a barra de tarefas ao invés de minimizar normalmente.
        + Float Statistics in Progression: Abre uma caixa de informações especificando número de tentativas bem sucedidas, número de passes já utilizados, entre outros. É muito importante pois através desta opção pode se saber do andamento do processo.
      * Snap Shots:
        + Enable Snap Shots: Libera o uso dos snap shots (ou instantâneos). Consistem em arquivos de configuração .ini onde estarão salvas as configurações específicas para o site alvo. Este arquivo também pode denominar onde na wordlist o processo parou em seu último uso, porém o SentryMBA possui essa função nativamente. Suas opções são:
        + Load Settings from Snap Shot; e
        + Save Settings to Snap Shot: As duas envolvem carregar um arquivo de configuração já salvo anteriormente, e salvar um arquivo novo (ou subscrever o mesmo arquivo), respectivamente.
    - Aba HTTP Header:

Nesta aba é configurado como o SentryMBA irá se comportar diante da requisição feita para o site alvo. Podemos ver as opções Request Header, que é onde se define o Header que o SentryMBA irá enviar para o site alvo, os User Agents, que são as versões de navegadores que serão interpretadas pelo alvo, o Request Method (ou método de requisição), que irá definir a metodologia usada, entre outros. Nesta aba, é preciso já ter realizado a captura de pacotes com o sniffer primeiro, para poder identificar como o pacote de solicitação de login se comporta.

* + - Aba Proxy Settings:

Onde é definido as configurações relativas ao proxy. Aqui podemos ver opções relativas a:

* + - * Banning and Reactivation: Possui opções relacionadas ao banimento e reativação dos proxies. Por exemplo, pode banir um proxy que não esteja alcançando o resultado esperado, ou até mesmo se ele conseguiu logar sucessivamente muitas vezes.
      * Waiting Window: Permite especificar, em minutos, quanto tempo o software irá levar para reiniciar, após todos os proxies terem sido banidos.
      * Banning Window: Na janela de tentativas por combo testado no último (minuto especificado, padrão 1) for maior que (minuto especificado, padrão 10) e os proxies que estiverem sobrando forem menos que (quantidade especificada, padrão 10), então esta opção bane todos os proxies.
      * Other Options
        + Do not use proxies: Não utilizará os proxies. Fortemente não recomendado, dependendo do uso que o programa terá.
    - Aba Fake Settings, ou Fake Pass Protection:

Possui configurações utilizadas na autenticação com o site alvo, forjando com diversas técnicas, a comunicação com o mesmo. Dentre suas opções é possível citar um exemplo, “Enables success keyword match retries”. Em alguns sites, quando a tentativa de login é falha, o site retorna uma resposta positiva, como se o login fosse sucedido. Esta opção irá prevenir justamente isso. Todas as outras opções nesta categoria agem desta forma, circulando métodos de proteção de login.

3. Soluções de segurança

Após ficar claro como um ataque é feito em um serviço de autenticação de contas e como é muito fácil ter acesso ao login de um usuário inserindo um banco de dados que pode ser criado ou até vazado de algum outro lugar, deve-se ter em mente uma forma para mitigar esse problema, um jeito em que mesmo que seja feito esse ataque o site se comporte de forma que possa bloquear ou até mesmo um jeito que esse ataque seja totalmente falho, tudo isso, para que possa ser garantido um dos pilares que é muito importante na área de segurança da informação, a confidencialidade.

É muito importante que uma organização pense nesse fator quando contratar um serviço de segurança da informação, a quebra da autenticidade pode causar prejuízos enormes, como perda de clientes, indenizações por vazar informações não autorizadas e principalmente a perda de crédito diante o mercado corporativo. Sendo assim, será demonstrado como pode-se proteger desse ataque automatizado feito através do software SentryMBA.

3.1 Método “Anti-Automação”

O que faz o ataque automatizado ser tão perigoso e eficiente é justamente porque o software consegue testar milhares de usuários e senhas em apenas alguns segundos conseguindo ser eficiente após milhares de solicitações de autenticação. Então, a primeira forma de mitigar será através do método de “anti-automação”, nele é possível pensar em formas que em cada solicitação, o site irá pedir um requisito a mais, que pode ser algo muito simples para uma pessoa física (humano) mas para a máquina pode ser praticamente impossível. Como por exemplo o “Captcha “.

Com a implementação do “Captcha” em um site, o objetivo é que após 3 solicitações falhas de autenticação, será necessário mais um requisito para se “logar”, além do usuário e senha. Será apresentado uma imagem aleatória com caracteres aleatórios onde o usuário terá que digitar o conteúdo em uma caixa de texto, caso as informações tiverem corretas, terá sucesso na autenticação. Esse método praticamente quebra o ataque automatizado, ele não conseguirá mais fazer muitas tentativas por segundo pois além do usuário e senha tem um terceiro requisito que não consta no banco de dados e é totalmente aleatório.

Apesar desse método se mostrar muito eficiente e é utilizado em muitos sites atualmente, ele pode se tornar falho com o avanço da inteligência artificial onde os computadores estão cada vez mais capazes de interpretar diferentes imagens como os que são gerados pelo “captcha”. Dessa forma, será explorado um pouco da segunda ideia que é a autenticação em dois fatores.

Com o mundo globalizado e como hoje em dia praticamente todos possuem um smartphone, a autenticação em dois fatores tem se mostrado um método muito eficiente de mitigação de ataques automatizados não autorizados. Nele, após a autenticação sendo feita com êxito, o site irá solicitar mais uma informação para poder acessar as informações que você deseja. Essa informação pode ser um “sms” enviado para um celular previamente cadastrado no sistema, um “token” que é gerado através de aplicativo ou código enviado para o e-mail. Com esse método, apesar de ser um pouco mais demorado par ao usuário final conseguir acesso ao site, fica bem difícil um sistema automatizado conseguir sucesso na invasão.

Deve-se realizar o monitoramento e registro de todas as ações de login dos usuários, o monitoramento permitirá uma ação imediata para o impedimento do ataque, a documentação dos registros das ações também poderá identificar ataques após análise dos registros e deve-se documentar para fins de auditoria e conformidade.

Uma forma de mitigação seria o rastreio de padrões durante o monitoramento do sistema, como utilização do mesmo cliente HTTP, dispositivo, User Agent. A identificação de padrões facilita o bloqueio dos ataques dos invasores.

Também é possível interromper um ataque limitando o número de contas que podem ser logadas de um endereço IP em um certo período. Com isso forçaria o atacante a no mínimo utilizar uma tecnologia como uma VPN que se altera toda vez que um certo número de ataques é realizado, para camuflar o seu IP. Um efeito parecido seria limitar as tentativas de login por cliente HTTP. Se seu sistema não possuir este tipo de segurança em funcionamento, o atacante pode tentar invadir o sistema indefinidamente.

Uma das formas para bloquear este ataque é por meio de ferramentas de inteligência de ameaça, normalmente é usado o Web Application Firewall (WAF), segundo a OWASP (2016), um WAF funciona como um firewall para aplicativos HTTP, ele aplica um conjunto de regras na comunicação com o site, ou seja, é realizado uma análise nas solicitações de GET e POST enviados por HTTP ou HTTPS e aplica regras configuradas para detectar e filtrar o tráfego malicioso da Web, e assim bloqueando a comunicação caso seja detectado que o pacote é malicioso.

Também é possível utilizar biblioteca de footprint, seria um modo de combinação de dados que não podem ser duplicados em outro lugar. Usando essas bibliotecas é possível verificar as semelhanças entre um usuário e grande fatias de tráfego, podendo criar padrões para que seja bloqueado caso tenha suspeita que não seja o usuário.

CONSIDERAÇÕES FINAIS

Ao longo do trabalho, foi-se verificado a possibilidade de um sistema mais seguro. Para isso, foram realizadas tentativas de ataque em um ambiente real, onde foram capturadas contas e as etapas de segurança que envolvem este processo. A partir disso, foram explorados os métodos que são utilizados para realizar a proteção dessas contas.

REFERÊNCIAS

ABBAS, R. **Backpropagation Networks Prototype for Offline Signature Verification.** RMIT, 1994. Disponível em: <https://s3.amazonaws.com/academia.edu.documents/31072265/10.1.1.48.921.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1558706518&Signature=ZWOQz5FMPnwC2d%2FddMavQnWKdLk%3D&response-content-disposition=inline%3B%20filename%3DBackpropagation\_Networks\_prototype\_for\_o.pdf>. Acesso em: 19 de maio de 2019.

BONNEAU, J et al. **The Quest to Replace Passwords:** A Framework for Comparative Evaluation of Web Authentication Schemes. 2012 IEEE Symposium on Security and Privacy, 2012. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6234436>. Acesso em: 16 de maio de 2019.

BOYD, S.W.; KEROMYTIS, A. D. **SQLrand:** Preventing SQL Injection Attacks. Columbia University, 2004. Disponível em: <http://web1.cs.columbia.edu/~angelos/Papers/sqlrand.pdf>. Acesso em: 15 de abril de 2019.

BRASSARD, G. **Advances in Cryptology** - CRYPTO'89 Heidelberg, Alemanha: LNCS, 1995. volume 435.

CRISTOFARO, E et al. **A Comparative Usability Study of Two-Factor Authentication**. Cornell University, 2013. Disponível em: <https://arxiv.org/pdf/1309.5344.pdf>. Acesso em: 20 de maio de 2019.

DHIMAN, M.; GLAZIER, W. **Automation Attacks at Scale -** Credential Exploitation. Stealth Security, Inc, 2017. Disponível em: <https://grehack.fr/data/2017/slides/GreHack17\_Automation\_Attacks\_at\_Scale\_paper.pdf>. Acesso em: 10 de abril de 2019.

FELTEN, E et al. **Password Management Strategies for Online Accounts**.Princeton University, 2006. Disponível em: <http://www.dphu.org/uploads/attachements/books/books\_3522\_0.pdf>. Acesso em: 17 de maio de 2019.

HALFOND, W. G. J.; ORSO, A.; VIEGAS, J. **A Classification of SQL Injection Attacks and Countermeasures**. Georgia Institute of Technology, 2006. Disponível em: <https://pdfs.semanticscholar.org/81a5/02b52485e52713ccab6d260f15871c2acdcb.pdf>. Acesso em: 19 de abril de 2019.

HEINEN, M et al. **Biometria Comportamental:** Pesquisa e desenvolvimento de um sistema de autenticação de usuários utilizando assinaturas manuscritas. Universidade Vale do Rio dos Sinos, 2004. Disponível em: <http://infocomp.dcc.ufla.br/index.php/INFOCOMP/article/view/72>. Acesso em: 18 de maio de 2019.

LAMPORT, L. **Password Authentication with Insecure Communication**. SRI International. Communications of the ACM, 1981. Disponível em: <http://merlot.usc.edu/cs530-s07/papers/Lamport81a.pdf>. Acesso em: 16 de maio de 2019.

LANGHEINRICH, M. **Privacy Invasions in Ubiquitous Computing**. Swiss Federal Institute of Technology, 2004. Disponível em: <https://uc.inf.usi.ch/wp-content/cache/mendeley-file-cache/fe35ed36-3d2c-3d85-9722-d956f47b96b7.pdf>. Acesso em: 18 de abril de 2019.

MORRIS, R; THOMPSON, K. **Password Security:** A Case History. Communications of the ACM, 2002. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.128.1635&rep=rep1&type=pdf>. Acesso em: 17 de maio de 2019.

OWASP. **The Ten Most Critical Web Application Security Risks**. 2017. Disponível em: <https://www.owasp.org/images/7/72/OWASP\_Top\_10-2017\_%28en%29.pdf.pdf>. Acesso em: 31 de março de 2019.

PESSOA, M. **Segurança em PHP**. Editora Novatec, 2007. Disponível em: <http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/254879.pdf>. Acesso em: 18 de março de 2019.

RENAUD, K. **Quantifying the quality of web authentication mechanisms**: a usability perspective, 2004. Disponível em: <<https://www.riverpublishers.com/journal/journal\_articles/RP\_Journal\_1540-9589\_322.pdf> Acesso em: 17 de maio de 2019.

RIVAS, G. **Web Application Security:** 7 Best Practices You Need to Know. GBadvisors, 2018. Disponível em: <https://www.gb-advisors.com/web-application-security-5-best-practices/>. Acesso em: 1 de abril de 2019.

SCHNEIER, B. **Customers, passwords, and Web sites**. IEEE Security & Privacy, Volume: 2, Edição: 4, 2004. Disponível em: <https://ieeexplore.ieee.org/document/1324609>. Acesso em: 17 de abril de 2019.

WEIDMAN, G. **Testes de Invasão:** Uma Introdução Prática ao Hacking. Santa **Terezinha (RS).** Editora Novatec, 2014.