FACULDADE DE TECNOLOGIA DE AMERICANA
Curso Superior de Tecnologia em Segurança da informação

Gabriel de Souza Alkimim
João Victor Dias Menegatti

Ataques de *SQL Injection* e Engenharia Social

Analisando e explicando as técnicas de invasão

Americana - SP

2020

FACULDADE DE TECNOLOGIA DE AMERICANA

Curso Superior de Tecnologia em Segurança da Informação

Gabriel de Souza Alkimim

João Victor Dias Menegatti

Ataques de SQL Injection e Engenharia Social

Analisando e explicando as técnicas de invasão

Trabalho de Conclusão de Curso desenvolvido em cumprimento à exigência curricular do Curso Superior de Tecnologia em Segurança da informação, sob a orientação do (a) Prof.(a) Esp. Marcus Vinicius Lahr Giraldi

Área de concentração: Segurança da Informação

Americana, SP
2020

FICHA CATALOGRÁFICA – Biblioteca Fatec Americana - CEETEPS
Dados Internacionais de Catalogação-na-fonte

A417a ALKMIM, Gabriel de Souza

Ataques de SQL Injection e engenharia social: analisando e explicando as técnicas de invasão. / Gabriel de Souza Alkmim, João Victor Dias Menegatti. – Americana, 2020.

32f.

Monografia (Curso Superior de Tecnologia em Segurança da Informação) - - Faculdade de Tecnologia de Americana – Centro Estadual de Educação Tecnológica Paula Souza

Orientador: Prof. Esp. Marcus Vinicius Lahr Giraldi

1 Engenharia social 2. Segurança em sistemas de informação 3. Banco de dados I. MENEGATTI, João Victor Dias II. GIRALDI, Marcus Vinicius Lahr III. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade de Tecnologia de Americana

CDU: 681.3.05

Gabriel de Souza Alkimim
João Victor Dias Menegatti

Ataques de SQL Injection e Engenharia Social

Analisando e explicando as técnicas de invasão

Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Segurança da Informação pelo CEETEPS/Faculdade de Tecnologia – FATEC/ Americana.

Área de concentração: Segurança da Informação.

Americana, 14 de Dezembro de 2020.

Banca Examinadora:

_____
Marcus Vinicius Lahr Giraldi (Presidente)
Especialista
Fatec Americana Ministro Ralph Biasi

_____
Renato Kraide Soffner (Membro)
Doutor
Fatec Americana Ministro Ralph Biasi

_____
Elton Rafael Mauricio da Silva Pereira (Membro)
Mestre
Fatec Americana Ministro Ralph Biasi

AGRADECIMENTOS

Em primeiro lugar agradecemos a todos os professores do curso de Segurança da Informação pela excelência da qualidade técnica. Aos nossos pais, que nos incentivaram a todo momento. Aos nossos amigos de sala, pelo companherismo e trocas de informações ao longo do curso.

RESUMO

O foco deste trabalho é compreender como funciona o mundo que engloba os testes de invasão, as ações dos *pentesters* que trabalham investigando os sistemas. Dessa forma, será possível compreender o funcionamento dos sistemas, levando também ao conhecimento relacionado as técnicas e conceitos utilizados pelos *pentesters* em seu trabalho, neste momento existirá uma abordagem mais específica a respeito da técnica de *Structured Query Language* (SQL) *Injection*, que é muito comum nos dias de hoje e se bem utilizada pode garantir diversas permissões a um atacante que esteja invadindo e explorando falhas existentes em um banco de dados, o que também pode vir a afetar todo o resto do sistema. Além disso, também será apresentada a técnica de Engenharia Social, que será utilizada para "manipular" os usuários ao invés de um sistema em si, e a partir dessa manipulação do usuário de um sistema, o atacante pode garantir vantagens para uma futura invasão que poderá ser feita a um determinado sistema. O conhecimento obtido a partir do estudo dessas técnicas poderá ser utilizado para compreender melhor qualquer sistema que virá a ser investigado, além de facilitar o entendimento do comportamento de um usuário, pode também melhorar o processo de melhoria da segurança do sistema em geral, visto que poderão ser corrigidas falhas do próprio sistema, e também falhas humanas que podem vir a ser evitadas a partir de um treinamento focado na análise da Engenharia Social. Essas técnicas serão demonstradas a partir de uma plataforma criada no Kali Linux, que é um dos ambientes mais comuns a serem utilizados para esse tipo de teste. Ao longo do projeto serão utilizadas ferramentas baseadas em Linux, além de alguns comandos utilizados nas mesmas.

Palavras Chave: *Pentesters*, invasão e manipulação.

ABSTRACT

The focus of this work is to understand how the world that encompasses penetration tests works, such as the actions of the pentesters who work investigating the systems. In this way, it will be possible to understand the functioning of the systems, also leading to related knowledge such as techniques and concepts used by the pentesters in their work, at this moment there will be a more specific approach regarding the Structured Query Language (SQL) Injection technique, which is very common nowadays and if well used, it can guarantee several flaws to an attacker who is invading and exploiting existing flaws in a database, which can also affect the rest of the system. In addition, there will also be and some Social Engineering techniques, which will be used to "manipulate" the users instead of a system itself, and from the manipulation of the user of a system, the attacker can guarantee advantages for a future invasion that can be made to a given system. The knowledge obtained from the study of these techniques can be used to better understand any system that will be investigated, in addition to facilitating the understanding of a user's behavior, it can also improve the process of improving the security of the system in general, since flaws in the system itself can be corrected, as well as human flaws that can be avoided through training focused on the analysis of Social Engineering. These techniques will be demonstrated using a platform created on Kali Linux, which is one of the most common environments used for this type of test. Throughout the project, Linux-based tools will be used, in addition to some commands used in them.

Keywords: *Pentesters, invasion and manipulation.*

SUMÁRIO

ÍNDICE DE FIGURAS

1. INTRODUÇÃO

A Segurança da Informação tornou-se um fator primordial, sendo uma das áreas mais em alta no mercado de trabalho, nos tempos atuais onde tudo e todos estão conectados, e diversas leis regulamentando o uso e armazenamento de dados, não há como negar que é necessário envolver a Segurança da Informação em tudo relacionado a tecnologia.

Por este motivo, o foco principal deste projeto é realizar um estudo e trazer informações sobre técnicas de invasão. Focando em técnicas que fazem uso de *Structured Query Language* (SQL) *Injection*, e um outro tipo de técnica que ao invés de ser utilizada contra um sistema é utilizada contra pessoas, que é conhecida como Engenharia Social

Dessa forma, este projeto poderá ser utilizado para conscientizar o usuário sobre como um sistema funciona, seus riscos e quais cuidados deverão ser tomados. Não só durante o uso do computador ou de algum sistema, mas a todo momento, levando em consideração que a Engenharia Social pode ser aplicada por um atacante a qualquer momento.

Levando em consideração que mesmo que o sistema ou a rede da empresa tenham as melhores tecnologias, as melhores técnicas de defesa e os melhores especialistas em Segurança da Informação ainda é possível que o sistema seja invadido intencionalmente ou não intencionalmente por meio da máquina de um dos funcionários comuns da empresa, seja por falta de conhecimento ou diversos outros motivos que podem estar envolvidos.

Ou seja, nenhum ambiente é totalmente seguro e o máximo que pode ser feito é tentar diminuir as ameaças que podem afetar o sistema ou minimizar as consequências.

2. DIFERENÇA ENTRE HACKERS ÉTICOS E CRACKERS

Com o passar dos anos, diversas categorias de hackers surgiram no mundo tecnológico, no entanto, ainda é comum que as pessoas categorizem de maneira errônea os ataques que acontecem, generalizando toda uma classe de pessoas que trabalham com segurança da informação. De acordo com a revista CIO, para se defender de um cracker deve-se pensar como um, desta forma nasce o termo *Ethical Hacking.* Um ataque de um cracker pode causar prejuízos gigantescos, podendo causar roubo de informações, prejuízo a imagem e paralisação dos serviços. Por conta desses riscos, empresas de várias áreas tem procurado por profissionais e empresas especializadas em *Ethical Hacking*. “Segundo Roger A. Grimes a definição de hacker ético é um profissional de segurança da informação especializado em *Offensive Security*, ou seja, na parte da segurança cibernética que é mais focada em processos de identificação de vulnerabilidades e, consequentemente, no desenvolvimento de métodos de proteção. O que ele faz? Em linhas gerais, ele é pago para tentar invadir sistemas, para detectar vulnerabilidades. Sua função é encontrar vulnerabilidades de segurança que um hacker malicioso poderia potencialmente explorar. Para tanto, precisa desenvolver habilidades em técnicas de penetração de sistemas”. Nos próximos subcapítulos serão introduzidas algumas categorias de “hackers”. (GRIMES, 2019)

2.1. Hacking ético

Em seu livro, James Broad descreve um hacker ético como um *pentester* profissional que ataca os sistemas em nome do proprietário do sistema ou da empresa proprietária do sistema de informação. O Hacking ético será sinônimo de Teste de Invasão. (BROAD, 2014, p.20)

2.2. White Hat

Segundo James Broad, *White Hat* (chapéu branco) é uma giria para um hacker ético ou um profissional da área de segurança de computadores, especializado em metodologias para melhoria da segurança dos sistemas de informação. (BROAD, 2014, p.20)

2.3. Grey Hat

Em seu livro, James Broad apresenta o termo *Grey Hat* (chapéu cinza), que refere-se a um especialista da área técnica que fica entre a linha que separa os *White Hats* dos *Black Hats*. Esses individuos normalmente tentam passar pelos recursos de segurança de um sistema de informação sem ter permissão, não para obter lucros, mas para informar os pontos fracos descobertos aos administradores do sistema. Os *Grey Hats* normalmente não têm permissão para testar os sistemas, porém, em geral, não estão atrás de lucros financeiros pessoais. (BROAD, 2014, p.21)

2.4. Black Hat

Segundo James Broad, *Black Hat* (chapéu preto) é um termo que identifica um individuo que usa técnicas para passar pela segurança dos sistemas sem ter permissão, para cometer crimes de cibernéticos. Os *pentesters* com frequência usam as técnicas utilizadas pelos *Black Hats* a fim de simular esses indivíduos ao conduzir exercícios ou testes autorizados. Os *Black Hats* conduzem suas atividades sem ter permissão e de forma ilegal. (BROAD, 2014, p.20)

3. CICLO DE INVASÃO

Os testes de invasão podem ser feitos por uma equipe interna de profissionais de Tecnologia da Informação (TI) de uma determinada organização, ou então por alguma empresa e(ou) pessoa terceirizada especializada nesse tipo de procedimento.

Por exemplo, caso um profissional terceirizado seja contratado o escopo de testes muda um pouco, e a primeira prioridade passa a ser o levantamento do máximo de informação sobre a empresa. Após isso, ele costuma procurar dados específicos da infraestrutura da empresa, como qual o tipo de rede, os softwares que são utilizados e como as pessoas operam.

Em seguida, é executado o escaneamento da rede, onde o profissional realizará uma vasta varredura em busca de vulnerabilidades, visando encontrar uma brecha para executar a invasão e efetuar o “roubo” dos dados. No final do processo, ele fará o possível para limpar todos os rastros e pistas deixados para trás durante o processo de invasão, além disso, ele deve montar um relatório sobre os resultados descobertos, esse relatório deve conter informações do tipo: quais brechas foram encontradas, como foi feita a invasão, quais dados ele conseguiu acesso, quais os danos que poderiam ter sido causados a empresa, etc.

A partir desse relatório fica mais fácil analisar a situação da infraestrutura da empresa, facilitando o planejamento para aprimorar a mesma. Dessa forma é possível corrigir falhas existentes podendo amenizar ou até mesmo evitar as invasões.

3.1. Passo a passo do ciclo de invasão

Segundo James Broad, existem diversos modelos diferentes de ciclo de vida dos testes de invasão, e o que será abordado aqui é a metodologia e o ciclo de vida definido e utilizado pelo programa EC CEH (EC-*Council Certified Ethical Hacker*). Esse processo possui cinco fases, que são: Fase de reconhecimento, *scanning*, obtenção de acesso, preservação do acesso e ocultação das pistas. Essas fases são basicamente um “plano de orientação” ao *pentester*.

3.1.1. *“Fase 1: Reconhecimento*

Essa fase tem como foco aprender absolutamente todas as informações sobre a rede e a empresa que serão o alvo do processo. Isso é feito por meio de pesquisas na internet, e também a partir de *scans* passivos nas conexões disponíveis

da rede. Durante esse processo o *pentester* não penetra realmente o sistema de defesa da rede, porém, identifica e documenta o máximo de informações a respeito do alvo.“ (BROAD, 2014, p.119)

3.1.2. *“Fase 2: Scanning*

O *pentester* utilizará as informações obtidas durante a primeira fase para iniciar o *scanning* da rede e do sistema alvo. Ao utilizar ferramentas nessa fase, será possível ter uma melhor definição da rede e da infraestrutura do sistema de informação que serão o alvo da exploração. As informações obtidas nessa fase poderão também ser utilizados durante o processo de exploração das falhas.” (BROAD, 2014, p.120)

3.1.3. *“Fase 3: Exploração de falhas*

O propósito dessa fase é invadir um sistema-alvo, a partir de vulnerabilidades que foram detectadas na fase anterior, com o objetivo de conseguir informações existentes no sistema, sem que o acesso seja notado. Como por exemplo: acessar remotamente uma máquina sem a necessidade de autenticação, através de login e senha ou por meio de tentativas de autenticação com senhas padrão em determinados sistemas.” (BROAD, 2014, p.121)

3.1.4. *“Fase 4: Preservação do acesso*

Essa fase ocorrerá após a execução bem-sucedida na exploração de falhas feita durante a fase 3, o *pentester* deve deixar *backdoors* e *rootkits* no sistema alvo, para que seja possível realizar novamente o acesso sempre que necessário. *backdoors* e *rootkits* são ferramentas que servem como uma porta de fácil acesso nos sistemas invadidos, também podem ser descritos como uma entrada secreta, pois, grande parte das vezes eles estão ocultos no sistema.” (BROAD, 2014, p.121)

3.1.5. *“Fase 5: Geração de relatórios*

Durante a última fase, o *pentester* deverá criar relatórios detalhados para explicar cada passo executado no processo de *hacking*, as vulnerabilidades exploradas e os sistemas que foram comprometidos. Além disso, em diversos casos um ou mais membros da equipe responsável pelo sistema alvo são responsáveis por analisar esse relatório, com o objetivo de compreender as vulnerabilidades existentes

no sistema.” (BROAD, 2014, p.121)

4. INJEÇÃO SQL

De acordo com o site PHP, um ataque de injeção SQL significa que será inserido uma consulta SQL por meio dos dados de entrada do cliente para o aplicativo. Uma pratica de injeção SQL bem-sucedida pode garantir diversas permissões ao atacante que está explorando as falhas de um banco de dados, como por exemplo: dados confidenciais, modificação desses dados, executar operações de administrador (como desligar o banco de dados) e em alguns casos, emitir comandos para o sistema operacional, ou seja, a injeção de SQL pode afetar não só o banco de dados, mas também pode afetar o sistema operacional da máquina/servidor que está rodando esse banco de dados. (PHP, 2008)

4.1. Modelagem de Ameaças:

Os ataques de injeção de SQL fazem com que os invasores usem identidades falsas, dessa forma adulteram os dados existentes, causam problemas de repúdio, como anulação de transações ou alteração de saldos, permitem a exibição completa de todas as informações do sistema, quebram os dados ou os tornam inutilizados de outra forma e tornam-se administradores do servidor.

A injeção de SQL é muito comum com aplicativos *Hypertext Preprocessor* (PHP) e *Active Server Pages* (ASP) por terem prevalência de interfaces funcionais mais antigas. Devido à natureza das interfaces programáticas disponíveis, os aplicativos *Java™ 2 Platform, Enterprise Edition* (J2EE) e ASP.NET são menos propensos a serem explorados facilmente por injeções de SQL.

A gravidade e os estragos causados pela injeção de SQL é limitada pela habilidade e imaginação do invasor e, em menor extensão, pelas contramedidas de defesa do ambiente alvo, como conexões de baixo privilégio com o servidor de banco de dados e outras medidas de segurança. Em geral, deve considerar a injeção de SQL uma ameaça muito critica.

4.2. Fatores de risco:

A injeção de SQL se tornou um problema comum em sites baseados em banco de dados. A falha pode ser facilmente detectada e explorada, dessa forma, qualquer site ou pacote de software com uma base mínima de usuários provavelmente estará sujeito a tentativas de ataque desse tipo. Por isso, é importante que as todas

empresas foquem em explorar, e consertar pelo menos as falhas mais simples de serem encontradas, dessa forma o site já se tornará mais seguro, visto que não será vítima de ataques que podem ser executados mais facilmente.

Essencialmente, o ataque é realizado colocando um metacaractere na entrada de dados para, em seguida, colocar os comandos SQL que antes não existiam no plano de controle. Essa falha depende do fato de que o SQL não faz distinção real entre os planos de controle e de dados.

4.3. Exemplos

“O código C # a seguir constrói e executa dinamicamente uma consulta SQL que procura itens que correspondam a um nome especificado. A consulta restringe os itens exibidos para aqueles em que o proprietário corresponde ao nome de usuário do usuário autenticado no momento.

```
...
string userName = ctx.getAuthenticatedUserName();
string query = "SELECT * FROM items WHERE owner = '"'
        + userName + "' AND itemname = '"
        + ItemName.Text + "'";
sda = new SqlDataAdapter(query, conn);
DataTable dt = new DataTable();
sda.Fill(dt);
...
```

A consulta que este código pretende executar é a seguinte:

```
SELECT * FROM items
WHERE owner =
AND itemname = ;
```

No entanto, como a consulta é construída dinamicamente pela concatenação de uma *string* de consulta de base constante e uma *string* de entrada do usuário, a consulta só se comporta corretamente se o itemName não contiver um caractere de aspas simples. Se um invasor com o nome de usuário wiley inserir a *string* "name' OR 'a'='a"para itemName, a consulta será a seguinte:

```
SELECT * FROM items
WHERE owner = 'wiley'
AND itemname = 'name' OR 'a'='a';
```

A adição da condição OR 'a'='a' faz com que a cláusula *where* sempre seja avaliada como verdadeira, de modo que a consulta se torna logicamente equivalente à consulta muito mais simples:

```
SELECT * FROM items;
```

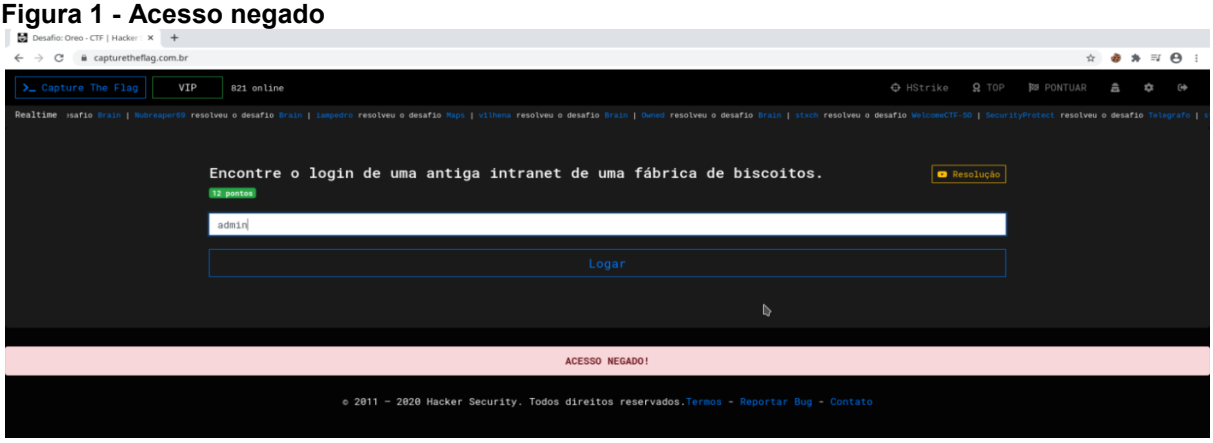
Essa simplificação da consulta permite que o invasor ignore o requisito de que a consulta retorne apenas itens pertencentes ao usuário autenticado; a consulta agora retorna todas as entradas armazenadas na tabela de itens, independentemente do proprietário especificado.” (OWASP, 2020)

4.4. Exemplos práticos:

Para esse exemplo utilizamos como ambiente de teste um site onde o foco é testar técnicas de invasão, ou seja, tínhamos permissão para realizar todos os testes.

Na Figura 1 tenta-se fazer o acesso no sistema sem saber o usuário correto.

Figura 1 - Acesso negado



Fonte: Alkimim, Menegatti (2020).

Na Figura 2 é exibido o editor de *cookies* que será utilizado e os *cookies* atuais do site.

Figura 2 - *Cookie editor*

Fonte: Alkimim, Menegatti (2020).

Na Figura 3 é inserido a condição para se ter um "*true*" no banco de dados.

Figura 3 - Inserção de condição

Fonte: Alkimim, Menegatti (2020).

Na Figura 4, após inserir a condição o site exibe um *cookie* "*admin*".

Figura 4 - Alterando o *cookie*

Fonte: Alkimim, Menegatti (2020).

Na Figura 5 o *cookie* “*admin*” é alterado para *true*.

Figura 5 - *Cookie* definido como *true*

Fonte: Alkimim, Menegatti (2020).

Na Figura 6 depois de definir o *cookie* “*admin*” como *true* e tentar fazer o acesso como *admin* o acesso é liberado.

Figura 6 - Acesso garantido

Fonte: Alkimim, Menegatti (2020).

Essa foi uma forma de se aproveitar da vulnerabilidade a SQL *Injection* do site alvo, mas também existem outras formas de se aproveitar dessa vulnerabilidade, outro exemplo que será apresentado se trata de uma ferramenta.

Existem ferramentas de código aberto, com foco em testes de penetração, elas podem automatizar o processo de detecção e exploração de falhas de injeção SQL. No próximo exemplo será utilizada uma ferramenta automatizada de injeção SQL, chamada SQLMap.

Para executar o SQLMAP é necessário utilizar um parâmetro *GET*, por exemplo, “www.site.com/index.php?id=1”, após isso é necessário explorar o site até

encontrar uma opção como é exemplificado na Figura 7.

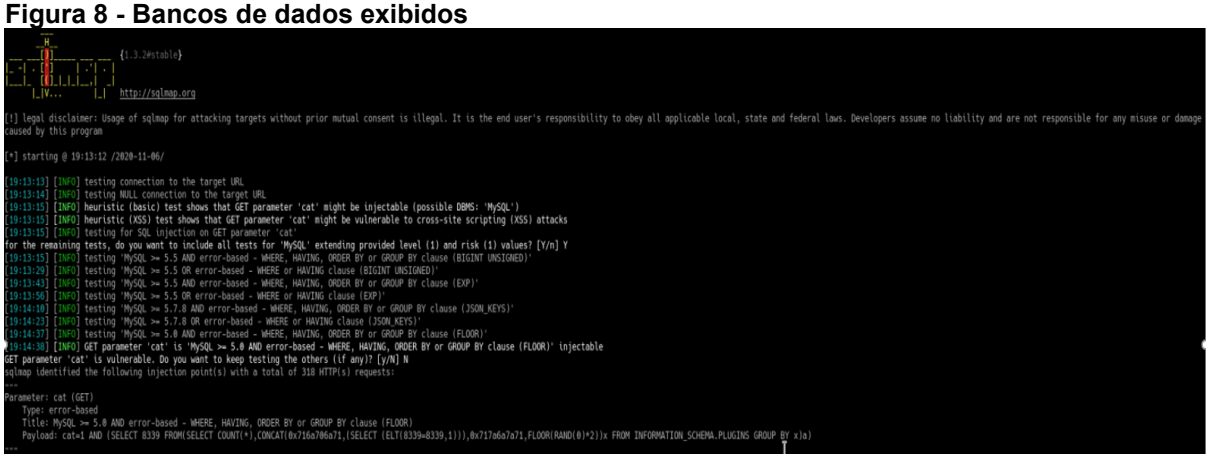
http://testphp.vulnweb.com/listproducts.php?cat=1

Figura 7 - SQL Map

Fonte: Alkimim, Menegatti (2020).

Durante a execução dos comandos, o SQLMAP já detecta o banco de dados do site, que nesse caso é MySQL, após isso aparece uma questão que busca saber se o usuário do comando deseja pular a verificação para outros tipos de banco de dados ou se deseja realizar um teste mais específico sobre o banco de dados detectado. Assim como é exemplificado na Figura 8.

Figura 8 - Bancos de dados exibidos



Fonte: Alkimim, Menegatti (2020).

Na Figura 9 são listados os bancos de dados, o próximo passo é extrair as tabelas do banco de dados acuart.

Figura 9 - Bancos de dados exibidos

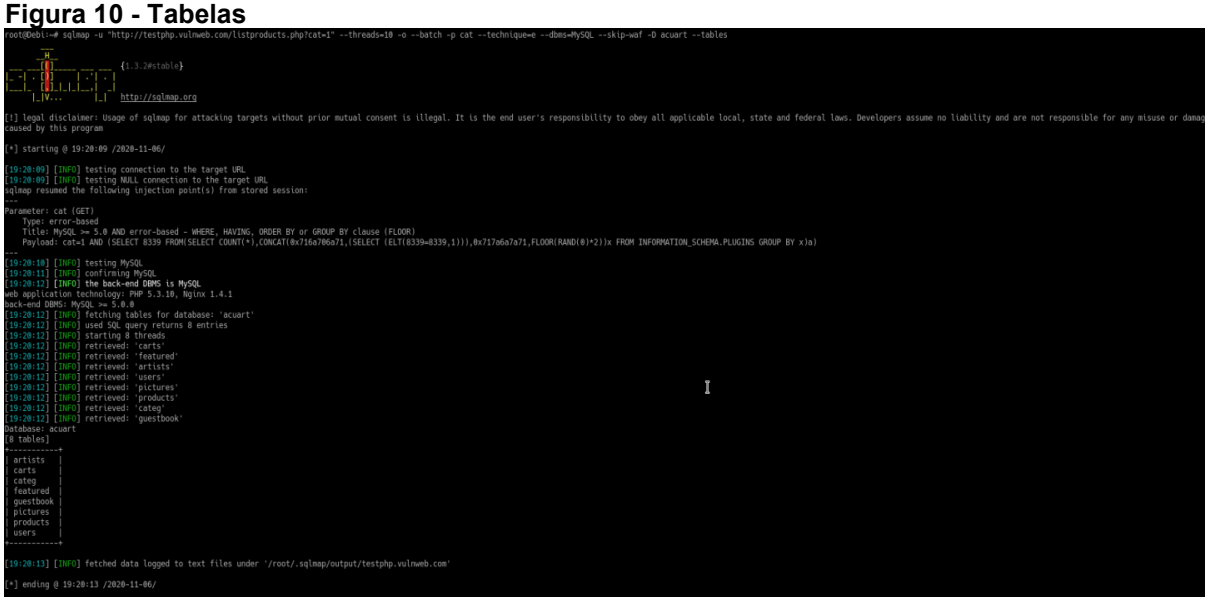
Fonte: Alkimim, Menegatti (2020).

[] acuart

[] information_schema

Na Figura 10, como resultado são extraídas 8 tabelas do banco de dados acuart.

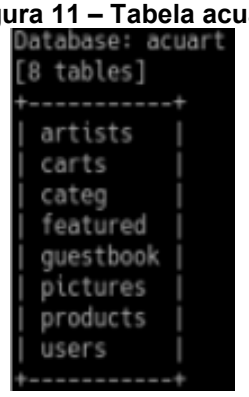
Figura 10 - Tabelas



Fonte: Alkimim, Menegatti (2020).

Na Figura 11 é possível ver mais detalhadamente as tabelas extraídas.

Figura 11 – Tabela acuart



Fonte: Alkimim, Menegatti (2020).

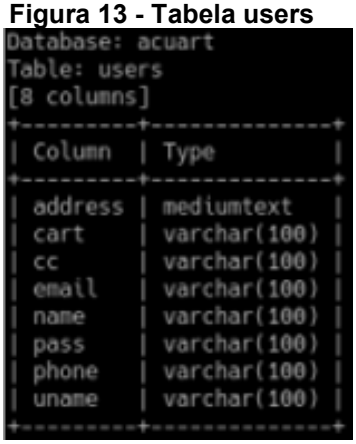
Na Figura 12 serão extraídas todas as colunas existentes na tabela users.

Figura 12 - Tabela

Fonte: Alkimim, Menegatti (2020).

Na Figura 13 é possível ver mais detalhadamente as tabelas extraídas.

Figura 13 - Tabela users



Fonte: Alkimim, Menegatti (2020).

Na Figura 14 será realizado o *DUMP*, ou seja, todas as informações cadastradas nas colunas da tabela *users* serão extraídas.

Figura 14 – Realização do *DUMP*

Fonte: Alkimim, Menegatti (2020).

Na Figura 15 é possível ver mais detalhadamente o resultado dos dados extraídos:

Figura 15 - Resultados *DUMP*

Fonte: Alkimim, Menegatti (2020).

5. Engenharia Social

A Engenharia Social está presente não só em ataques digitais, mas também fora deles, existem diversas pessoas que fazem o uso dessa técnica durante o cotidiano, de forma a explorar os dados de sua vítima mesmo fora do computador. A utilização dessa técnica de ataque digital é um fator consideravelmente importante nos dias atuais, e se for utilizada da maneira correta há uma probabilidade de aumentar as chances de sucesso em um ataque digital, visto que a partir da Engenharia Social os atacantes podem obter informações e dados muito valiosos que poderão ser utilizados a seu favor durante o ataque. Segundo Mitnick a Engenharia Social toma proveito do elo mais vulnerável na corrente de Segurança da Informação, esse elo que também é a parte mais importante, ou seja, os humanos. (MITNICK, 2003)

A Engenharia Social é uma técnica que funciona como uma arte de enganar ou manipular pessoas, persuadindo-as a passar informações pessoais que não deveriam ser compartilhadas, como por exemplo senhas e dados confidenciais. Essa técnica é muito utilizada por hackers de categoria *Black Hat*, ou seja, hackers mal-intencionados.

Segundo a Hacker Security existem dois tipos de Engenharia Social, a que é baseada em pessoas, e a que é baseada em computadores.

> “Engenharia Social Baseada em Pessoas:
> -Disfarces
> -Representações
> -Uso de *HelpDesk*
> -Baseadas em Computadores
> -E-mails falsos
> -Cavalos de troias anexados a e-mails
> -*Website* falso
>
> Engenharia Social Baseada em Computadores:
> -Trojans
> -E-mails falsos
> -*Websites* falsos

5.1. *Insider Attacks*

Insider é uma pessoa de dentro da empresa, geralmente algum funcionário ou ex-funcionario que tem acesso a informações confidenciais ou contas de acessos na rede da organização e querem fazer mal uso dessas informações.

5.2. Roubo de identidade:

De acordo com a Hacker Security, roubo de identidade é quando os criminosos utilizam as informações de outras pessoas para realizar ataques ou falsificações. Seja para obter lucro ou apenas informações sigilosas.

5.3. *Phishing Scam*

Essa forma de ataque é usada por hackers *Black Hat*, geralmente é utilizada com o intuíto de obter informações sigilosas, através de trojans, emails falsos ou páginas falsas.

5.4. *Vishing*

Segundo os analistas da Hacker Security essa técnica de ataque funciona como um “trote”, onde os criminosos ligam para a vítima e tentam obter informações, como por exemplo se passam por um atendente de banco solicitando dados sensíveis, dessa forma eles são capazes de efetuar um acesso não autorizado utilizando as informações da vítima para fazer compras, transferências ou até mesmo saques.

5.5. *Smishing*

Smishing é considerado pelo grupo Hacker Security como uma forma de ataque semelhante ao Phishing, porém essa técnica é feita através de mensagens SMS, onde geralmente a vítima recebe uma mensagem com uma solicitação para que uma ação “urgente” seja tomada, como por exemplo uma troca de senha.

5.6. Url *Obfuscation*

É uma técnica de “encurtar” *Uniform Resource Locator* (URL) para ficar mais simples, utilizada também por hackers *Black Hat* para esconder URLs maliciosas, dessa forma é possível enganar o usuário, fazendo com que ele acesse

um determinado link malicioso.

5.7. *Dumpster Diving*

É o termo utilizado para descrever uma técnica onde hackers vasculham o lixo de uma empresa, pois muitas empresas não dão a devida atenção a forma como descartam seu lixo, como por exemplo documentos, *Hard Disk* (HD), computadores, etc. Com esse método o hacker mal-intencionado pode encontrar muitas informações valiosas e até mesmo informações sigilosas que podem ajudá-lo a ter maior sucesso na sua busca por informações, garantindo a eles uma vantagem sobre a empresa e seus funcionários.” (HACKER SECURITY, 2018)

5.8. Persuasão

De acordo com a Hacker Security a Engenharia Social pode ser vista como uma técnica que mexe com o “psicológico” de uma pessoa, pois ela utiliza alguns métodos com o objetivo de persuadir alguma vítima a executar ações que irão beneficiar o atacante, alguns tipos de persuasão são: insinuação, personificação, conformidade e até mesmo velha amizade. Porém, independente da técnica de persuasão utilizada geralmente todas elas tem um objetivo em comum, garantir que o atacante obtenha uma vantagem, como por exemplo informações valiosas. (HACKER SECURITY, 2018)

Algumas contramedidas recomendadas pelo Hacker Security podem ajudar a prevenir esses ataques, por exemplo:

“- Mantenha-se protegido, não trabalhe em assuntos privados em locais públicos.

- Não passe informações importantes ou sigilosas por e-mail.

- Faça o descarte seguro de documentos.

- Utilize fechaduras e trancas de boa qualidade e comprovado nível de segurança.

- Mantenha bolsas e documentos pessoais em segurança.

- Não guarde suas senhas em documentos.

- Teste constantemente seus dispositivos de segurança, câmeras e detectores de - movimento.

- Mantenha-se atento aos engenheiros sociais.” (HACKER SECURITY,

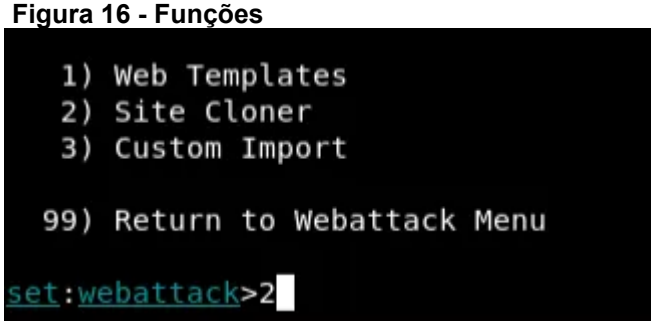
2018)

O exemplo apresentado a seguir trata-se de uma forma utilizada para clonar páginas *web*, fazendo com que a vítima que acessa o site clonado seja incentivada a informar seus dados na página falsa, imaginando que irá realizar o acesso a sua conta normalmente. Enquanto na verdade o atacante receberá esses dados que foram digitados pelo usuário, esse tipo de ataque é conhecido como Phishing.

Para realizar este teste será utilizada uma ferramenta chamada *Social-Enginner Toolkit* (Setoolkit), essa ferramenta foi criada e escrita pelo fundador da TrustedSec (Devon Kearns, Jim O'Gorman e Mati Aharoni). É uma ferramenta de código aberto escrita em Python, destinada a *pentesters* que desejam utilizar da engenharia social.

O Setoolkit é uma ferramenta que tem várias funções, e na Figura 16 será utilizada a função *site cloner:*

Figura 16 - Funções



Fonte: Alkimim, Menegatti (2020).

Após selecionar a função de clonar sites, é possível ver na Figura 17 que será solicitado o endereço do site que será clonado:

Figura 17 - Link a ser clonado

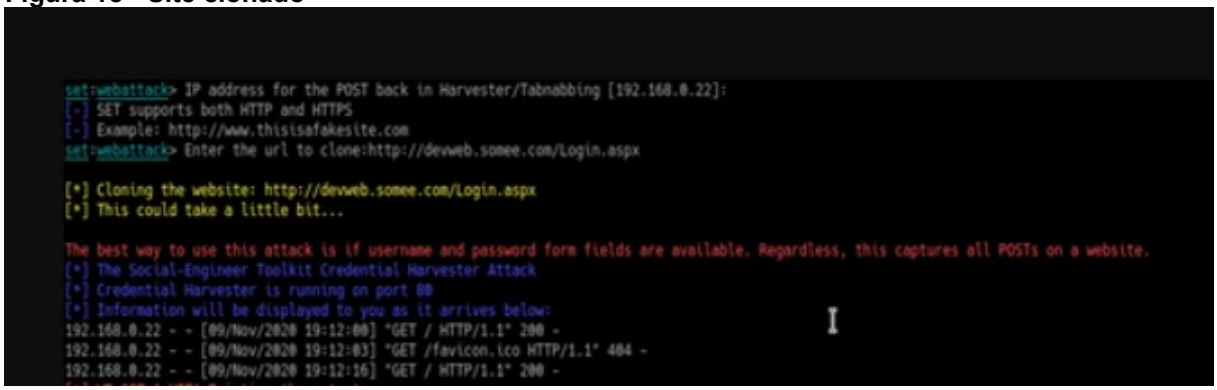
Fonte: Alkimim, Menegatti (2020).

Na Figura 18 será clonado um site de autoria própria, desenvolvido em aula.

http://devweb.somee.com/

A partir desse novo endereço, será mostrado uma tela onde o setoolkit aguarda qualquer interação do usuário no site clonado.

Figura 18 - Site clonado



Fonte: Alkimim, Menegatti (2020).

Este é um ataque interno, ou seja, o atacante e a vítima devem estar utilizando a mesma rede local, quando um usuário dessa rede acessar o site utilizando o link falso se ele não prestar atenção em alguns pontos que podem ajudar a identificar se o site é legítimo ou não pode ser que ele tenha suas informações roubadas, levando em consideração que qualquer informação digitada pela vítima será recebida pelo atacante diretamente pelo console da ferramenta Setoolkit, assim como na Figura 19, onde um usuário tentou realizar um login no site falso:

Figura 19 - Dados obtidos

Fonte: Alkimim, Menegatti (2020).

Na Figura 20 é possível ver mais detalhadamente a tela de login do site clonado, e por ser idêntica a tela de login do site original um usuário comum não conseguiria diferenciar as duas.

Figura 20 - Tela de login

Fonte: Alkimim, Menegatti (2020).

Com isso, qualquer usuário comum que acessar a página poderá observar que ela é idêntica a página original, no entanto, existem alguns aspectos a serem observados que podem deixar mais claro se a página é legitima ou não, por exemplo, é possível analisar o endereço do site para validar se não há algo de incomum, e também analisar o certificado *Secure Sockets Layer* (SSL) da página, visto que um site clonado provavelmente não terá um certificado. Caso contrário a vítima pode acabar tendo suas informações roubadas sem nem perceber.

6. CONSIDERAÇÕES FINAIS

Atrelado às questões acima citadas, pode-se considerar que por um bom tempo a tecnologia e a segurança dos sistemas continuarão evoluindo rapidamente, e junto delas também evolui a quantidade de pessoas mal-intencionadas e sua capacidade de quebrar a segurança desses sistemas. Diante disso o objetivo principal do trabalho foi apresentar algumas técnicas que invasores utilizam, para que dessa forma o usuário fique atento aos perigos que existem ao navegar na internet, podendo amenizar o prejuízo desses tipos de “golpes” e ataques, ou até mesmo podendo evitá-los.

Dessa forma, no cenário atual é necessário aumentar o nível de formação e(ou) instrução dada aos usuários, profissionais ou cidadãos em geral, pois a Segurança da Informação não é somente problema do profissional de TI, mas sim um problema de todos, afinal os ataques feitos por hackers não são feitos exclusivamente a grandes organizações, na verdade se tornou bem comum que usuários comuns caiam em “golpes” através de links ou algum outro tipo de mídia acessada via redes sociais. Um exemplo clássico que foi tratado durante o trabalho foi o uso da engenharia social que tem como foco o usuário, que é o elo mais fraco da tecnologia. Assim sendo, esta conscientização deve ser dada desde o início do ensino fundamental, iniciando uma cultura de abordagem clara, pois a cada dia a iniciação digital se dá pessoas mais jovens.

Certamente o assunto de ataques cibernéticos e Segurança da Informação requer um maior diálogo e consequentemente estudos mais aprofundados, com o intuito de passar ensinamentos no quesito de defesa de um sistema, mesmo sabendo que nenhum sistema está totalmente protegido, com algumas técnicas ou até mesmo o que pode vir a ser senso comum, como por exemplo não abrir *links* de *e-mails* desconhecidos, será possível minimizar ou evitar os impactos de um possível ataque.

Finalmente, muito ainda há de ser dito e construído, a Segurança da Informação é um desafio para todos. O foco desse trabalho era de disseminar o conhecimento sobre algumas técnicas de ataque utilizadas nos dias de hoje, e com isso ter ciência de tal, em prol da Segurança da Informação.

REFERÊNCIAS BIBLIOGRÁFICAS

Ataque de Engenharia Social. **Hacker Security**, 2018. <https://hackersec.com/ataque-de-engenharia-social/>. Acesso em: 16 ago. 2020.

BROAD, James; BINDNER, Andrew. Hacking com Kali Linux: Técnicas práticas para testes de invasão. 1ª edição. São Paulo: Novatec, 2014.

GRIMES, Roger. O que é hacking ético?. **CIO**, 2019. Disponível em: <https://cio.com.br/gestao/o-que-e-hacking-etico/>. Acesso em: 29 nov. 2020.

MITNICK, Kevin; SIMON, William. A Arte de Enganar – Ataques de Hackers: Controlando o Fator Humano na Segurança da Informação. São Paulo: Pearson Education do Brasil, 2003.

SQL Injection. **Owasp**, 2020. Disponível em: <https://owasp.org/www-community/attacks/SQL_Injection>. Acesso em: 10 out. 2020.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner. **Acunetix**, 2019. Disponível em: <http://testphp.vulnweb.com/listproducts.php?cat=1>. Acesso em: 23 ago. 2020.

Injeção de SQL. **PHP**, 2008. Disponível em: <https://www.php.net/manual/pt_BR/security.database.sql-injection.php#security.database.sql-injection>. Acesso em: 10 jul. 2020.

Injeção de SQL. **PHP**, 2008. Disponível em: <https://www.php.net/manual/pt_BR/security.database.sql-injection.php#security.database.sql-injection>. Acesso em: 10 jul. 2020.