ISTEC Instituto Superior de Tecnologias Avançadas do Porto

2025

Relatório Wargaming

BEATROOTER

RAFAEL BAPTISTA - Nº 2024134
DAVIDE FERREIRA - Nº 2024154
SAMUEL ROCHA - Nº 2024127
RODRIGO BALTAZAR - Nº 2024142
JOSÉ COUTO - Nº 2024026

ISTEC Instituto Superior de Tecnologias Avançadas do Porto

Índice

Introdução

No mundo atual, as principais **infraestruturas** de um país utilizam de alguma maneira computadores para o seu funcionamento. Sendo vital para o seu funcionamento, a **segurança digital** destas infraestruturas é essencial para a **segurança nacional**, e como consequência um **alvo prioritário** de eventuais atacantes. Com o objetivo de estudar e analisar a importância do **wargaming** no campo da **cibersegurança**, este relatório visa abordar o tema através da framework **Unified Kill Chain** e apresentar uma nova ferramenta de análise e preparação de ataques cibernéticos.

A ferramenta chamada **BeatRoot** é a nossa proposta para planear e armazenar informações coletadas durante as fases iniciais do Unified Kill Chain. Nela é possível criar um **mapa do sistema alvo** e transferir essas informações para outros atacantes, de modo a facilitar todo o **processo de ataque**.

Também utilizável para a **visualização** e **análise** dos próprios sistemas, o BeatRoot é uma ferramenta que auxilia administradores a protegerem contra **ameaças** e **ciberataques**.

Para além da ferramenta, também nos foi pedido assumir o papel de um **grupo de hackers** e desenvolver um **mindset** que irá guiar a equipa dentro do contexto de wargaming, levando em consideração a forma de operar, a **motivação** e o **modo de agir**. Por isso, este relatório inicia-se com o modo como pensamos e abordamos o wargaming a nível **estratégico** e **operacional**.

Escolha de alvos e motivação dos ataques – our Modus Operandi

A escolha do **alvo** é parte fundamental para o início de um **ciberataque**; sem um adversário não há ataque, e a definição clara e objetiva de quem é o adversário é o início de qualquer operação. A **motivação** da escolha — política, ideológica, militar ou financeira — deve levar em conta a existência de um **sistema**, **serviço**, **infraestrutura** ou **pessoas** concretas.

Motivações generalistas como lutar contra a **corrupção**, vencer a **guerra** ou derrubar determinado **regime** devem considerar que estrutura real conduz a esses objetivos. Isso implica compreender o funcionamento das **infraestruturas principais** do adversário e escolher um alvo que realmente cause **impacto** no objetivo final. Tomar controle e desabilitar qualquer serviço não implica gerar impacto real e ainda pode **alertar** o adversário sobre as intenções.

Uma vez selecionado um alvo capaz de causar dano significativo, toda a **motivação ideológica** e o motivo do ataque não devem interferir no modo de agir da **equipa técnica** responsável pela missão. O momento de decidir atacar já passou; agora a equipa foca apenas em **como** o ataque será conduzido da forma mais **eficiente**, utilizando todos os meios necessários para alcançar os **objetivos**.

Operar em equipa – our Modus Operandi

É muito comum na comunidade **hacker** a busca por **reconhecimento**. Muitos hackers realizam ataques puramente pelo **desafio** e para alcançar **fama** dentro da comunidade, o que pode levar ao **individualismo** e à **competitividade**. Esse comportamento prejudica a execução eficaz do objetivo, pois a **divisão de tarefas** em equipa acelera o cumprimento das metas e, em um contexto de **guerra cibernética**, o **tempo** é fator determinante para a vitória. Por isso, o **compartilhamento de informações** com a equipa de ataque é fundamental para um ataque **bem-sucedido**.

The Unified kill chain e Find&Share – our Modus Operandi

A escolha da **Unified Kill Chain** como framework base para os nossos ataques ocorre devido à sua **plasticidade** para diferentes tipos de operação. Para além do que está definido na framework, a nossa proposta é estabelecer, dentro das fases iniciais **Reconnaissance** e **Resource Development**, um componente de **compartilhamento de informações** em equipa que acelere o processo e facilite a identificação de **vulnerabilidades**.

Uma espécie de **framework dentro da framework**.
Nós a chamamos de **Find&Share**.

Figura 1 Find&Share

Search: A procura por vulnerabilidades, cada membro da equipa deve estar responsável por uma busca.

Find: Ao encontrar algo deve ser levado em consideração se é relevante ou não para o ataque.

Mapping: A informação encontrada deve ser inserida dentro do mapa do sistema adversário de modo coerente ao sistema.

Share: O mapa deve ser compartilhado e atualizado a cada nova descoberta relevante.

Exemplo de uso do Find&Share

Durante a preparação de um ataque, diversas **buscas** no sistema adversário são realizadas para coletar informações necessárias. Essas informações, quando **partilhadas** entre os membros da equipa de ataque, aceleram a fase de busca e tornam o ataque mais **eficiente**. Abaixo, um exemplo de uso do **Find&Share** em um ataque a um **servidor**:

Search: Um dos atacantes realiza uma busca no servidor e procura por **portas abertas** utilizando o **Nmap**. Para evitar ser detetado, evita gerar **ruído na rede**, tornando a busca mais lenta e levando horas para ser concluída.
O segundo atacante foca na busca dentro do serviço **SMB** que corre no servidor alvo. Com **SMBmap**, procura **pastas ocultas** com permissões mal configuradas de escrita e leitura.

Find: Os atacantes encontram diversas informações e selecionam as mais **relevantes**. No exemplo, o primeiro atacante percebe que a porta **44815** corre um **WordPress interno** sem criptografia.
O segundo atacante identifica uma **pasta compartilhada oculta** no serviço SMB com permissões de **escrita** para usuários guest.

Mapping: Os dois atacantes criam um **mapa do servidor** com as vulnerabilidades encontradas, registrando **portas abertas** relevantes e **pastas vulneráveis**, desenhando uma visão clara de como o servidor opera.

Share: Essa informação é **compartilhada** entre os atacantes, que decidem como **prosseguir**. Novos objetivos de Search são definidos com base nas novas informações, e o ciclo se repete até que o **ataque** seja finalmente realizado.

BeatRooter como solução

Com o objetivo de facilitar o processo **Find&Share**, desenvolvemos uma ferramenta de **mapeamento de vulnerabilidades**, o **BeatRooter**. No BeatRooter é possível criar um **mapa de ciberataque** completo, incluindo todas as informações relevantes e gerar um ficheiro simples de **partilhar** entre a equipa.

O BeatRooter conta com diversas **ferramentas de reconhecimento** que facilitam o processo de **search** e agilizam a **coleta de informações**.

Vejamos um exemplo de seu funcionamento em um **ataque**:

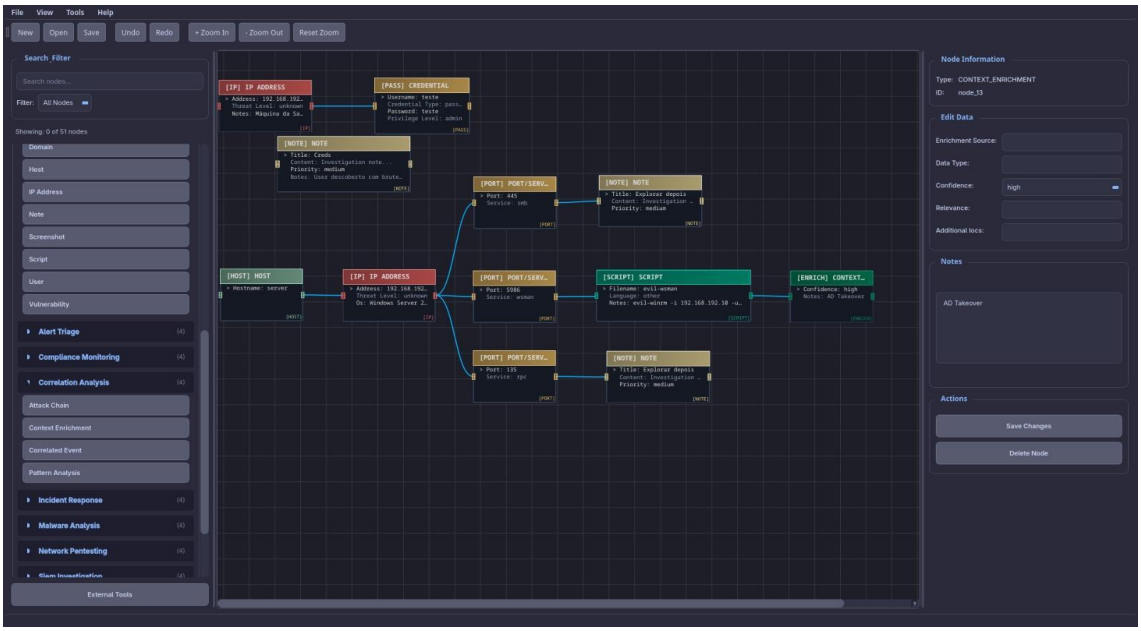


Figura 2 Exemplo de um mapa de ataque no BeatRooter

Neste exemplo, foi encontrado uma vulnerabilidade de uma conta de administrador em que a palavra-passe foi descoberta através de *brute force.*

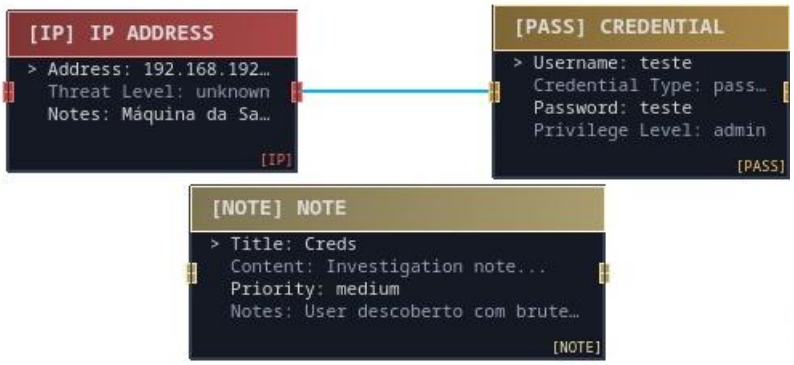


Figura 3 Usuário e senha descobertos e documentados.

Abaixo, o uso do privilégio de administrador permite o *Comand and control* do server através da porta 5986 no serviço wsman:

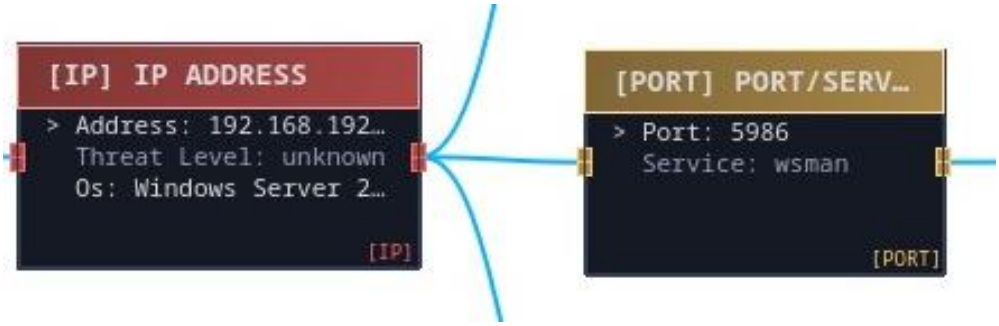


Figura 4 Em vermelho o servidor alvo e seu IP, em amarelo a porta e o serviço encontrado.

A entrega do ataque através do serviço é demonstrada nesses nodes:

Figura 5 Nota de como foi executado o ataque e resultado.

Neste exemplo simples, a informação de como executar um *Command and Control* no servidor é facilmente explicada neste diagrama e pode ser compartilhada com outros membros da equipa para enfim a execução dos objetivos do ataque.

Pentest x Realidade

Para prever e mitigar ataques, muitas organizações recorrem a **testes de intrusão (Pentests)** como parte de seu processo de melhoria contínua da **segurança interna**. Esses testes são conduzidos por uma equipa técnica habilitada que opera dentro de limites previamente estabelecidos pela organização, limites que definem o **escopo**, as **técnicas permitidas** e os **ativos** que podem ser explorados. Embora tais restrições contribuam para a segurança e para o controle de riscos operacionais, elas reduzem o **realismo da simulação**: quanto mais estreito o escopo, menor a correspondência com o comportamento de um **atacante real**, que não está sujeito a regras, contratos ou obrigações éticas.

Tendo isso em mente, ao planear um ataque, é provável que o defensor já tenha simulado e reforçado os **vetores mais óbvios** dentro do seu próprio modelo de segurança. Assim, seguir exatamente o caminho esperado diminui a probabilidade de sucesso. O **pensamento ofensivo**, portanto, deve buscar soluções “fora da caixa”, aproximando-se do que alguns autores chamam de **Transcendência Gödeliana**: a capacidade de ultrapassar os limites de um **sistema formal**, como políticas rígidas, doutrinas pré-estabelecidas ou padrões previsíveis, para conceber abordagens **originais** e **inesperadas**.

Essa criatividade é possível porque, além das limitações estruturais de qualquer metodologia de teste, muitos Pentests são deliberadamente **restringidos** para proteger **informações sensíveis**, evitar **impactos operacionais** e resguardar **interesses internos** da organização. Como resultado, parte significativa dos possíveis **vetores de ataque** permanece fora da simulação, ampliando ainda mais a diferença entre o teste controlado e a realidade de uma **operação ofensiva genuína**.

Apêndice A - Política de Segurança da Informação

1. OBJETIVO DO DOCUMENTO

O objetivo deste documento é estabelecer a política de segurança da informação da "Beatrooter", de forma a assegurar que os requisitos de confidencialidade, integridade e disponibilidade da informação relacionada com investigação, desenvolvimento, simulações de ataque, dados de clientes e operações críticas sejam adequadamente protegidos. A política também visa assegurar o cumprimento das obrigações legais, regulamentares e contratuais e a melhoria contínua do Sistema de Gestão de Segurança da Informação (SGSI).

1.2 ÂMBITO DE APLICAÇÃO

Esta política aplica-se obrigatoriamente a:

- Sistemas internos de desenvolvimento;
- Infraestrutura que suporta as plataformas de simulação;
- Equipamentos e dispositivos utilizados pelos colaboradores;
- Informação armazenada, processada ou transmitida pela Beatrooter;
- Fornecedores e terceiros com acesso a dados ou sistemas críticos.

1.3 VIGÊNCIA

Qualquer revisão deste documento entra em vigor imediatamente após sua publicação, substituindo versões anteriores. Alterações ou impossibilidades técnicas de aplicação devem ser reportadas ao Responsável de Segurança da Informação (RSI) para avaliação e implementação de medidas corretivas.

1.4 REVISÃO E AVALIAÇÃO

Este documento será revisto anualmente ou sempre que necessário, incluindo:

- Alterações significativas nos sistemas ou operações;
- Identificação de vulnerabilidades críticas;
- Novos requisitos legais ou contratuais.

Responsabilidade da revisão: Responsável de Segurança da Informação (RSI)
Aprovação do documento: Direção da Beatrooter

1.5 DOCUMENTOS DE REFERÊNCIA

- Normas ISO/IEC 27001:2022
- Políticas internas de segurança da informação da “Beatrooter”;
- Legislação e regulamentação aplicável à proteção de dados e segurança da informação;
- Procedimentos de desenvolvimento seguro e gestão de incidentes.

2. POLÍTICA DE SEGURANÇA DA INFORMAÇÃO

A proteção, tratamento, salvaguarda e transmissão de informação crítica é uma prioridade da Beatrooter, sendo essencial para o sucesso das operações, proteção de dados de clientes e continuidade do negócio. A perda ou exposição indevida de informação pode causar danos legais, financeiros e reputacionais.

O SGSI da Beatrooter é constituído por políticas, processos e procedimentos que visam:

- Proteger a informação e os ativos de suporte (sistemas, redes, infraestrutura);
- Avaliar continuamente riscos e implementar medidas adequadas;
- Garantir conformidade com legislação, regulamentação e contratos;
- Promover cultura interna de segurança e formação contínua.

2.1 PRINCÍPIOS

A política da “Beatrooter” garante que:

- A informação crítica está protegida contra acessos não autorizados;
- A confidencialidade, integridade e disponibilidade da informação são asseguradas;
- Planos de continuidade de negócio e recuperação de dados são implementados e testados regularmente;
- Todas as quebras de segurança são investigadas de forma estruturada;
- Os colaboradores conhecem e cumprem suas responsabilidades de segurança.

2.2 OBJETIVOS

Os principais objetivos do SGSI incluem:

- Garantir que todos os colaboradores compreendem suas responsabilidades e cumprem políticas de segurança;
- Estabelecer controles de acesso baseados no princípio do menor privilégio, com autenticação multifator;
- Classificar a informação em quatro níveis (Pública, Interna, Restrita, Crítica) e definir requisitos para armazenamento, transmissão e encriptação;
- Garantir segurança física e lógica, segregação de redes e monitorização contínua de sistemas críticos;
- Integrar práticas de desenvolvimento seguro, incluindo revisão de código, testes de vulnerabilidade e ambientes isolados para simulações;

- Gerir incidentes de forma estruturada, com detecção, registro, análise, resposta e relatórios pós-incidente;

- Garantir continuidade de negócio com planos documentados, backups cifrados e testes regulares;

- Avaliar riscos de fornecedores e monitorizar cumprimento de cláusulas de segurança contratual;

- Monitorizar indicadores de desempenho, incluindo incidentes registados e resolvidos, taxa de formação e tempo médio de resposta.

2.3 RESPONSABILIDADES

- **Responsável de Segurança da Informação (RSI):** coordena o SGSI, aprova políticas e supervisiona auditorias;

- **Equipa Técnica de Segurança:** gere acessos, monitorização, ferramentas defensivas e resposta a incidentes;

- **Equipa de Desenvolvimento:** assegura práticas de programação segura e gestão de vulnerabilidades;

- **Todos os Colaboradores:** cumprem políticas, reportam incidentes e participam em formação periódica.

Todos devem garantir a implementação, manutenção e melhoria contínua do SGSI, reportando prontamente qualquer incidente ou violação de segurança.

3. INDICADORES DE DESEMPENHO E MONITORIZAÇÃO

A eficácia das políticas é monitorizada por:

- Número de incidentes registados e resolvidos;
- Taxa de conclusão de formação em segurança;
- Tempo médio de resposta a incidentes;
- Resultados de auditorias internas e externas;
- Nível de conformidade com critérios de segurança nos projetos de desenvolvimento.

3.1 PERIODICIDADE DE MONITORIZAÇÃO

Os indicadores definidos na secção 3 são analisados com a seguinte periodicidade:

- Incidentes registados e resolvidos: mensal
- Taxa de formação obrigatória: trimestral
- Tempo médio de resposta a incidentes: mensal
- Resultados de auditorias internas: anual
- Conformidade de segurança em projetos de desenvolvimento: por projeto

Relatórios de desempenho são apresentados pela Equipa Técnica de Segurança ao RSI e à Direção.

4. REVISÃO E MELHORIA CONTÍNUA

O SGSI segue o ciclo de melhoria contínua **Planear – Executar – Verificar – Actuar**, assegurando que os mecanismos de segurança são relevantes, adequados e eficazes. Revisões ocorrem sempre que houver alterações significativas, vulnerabilidades críticas ou novos requisitos legais.

4.1 AUDITORIAS

Para garantir a eficácia do SGSI, são realizadas auditorias internas anuais e auditorias externas sempre que necessário.

As auditorias avaliam:

- Cumprimento de políticas e procedimentos
- Controlo de acessos
- Gestão de incidentes
- Conformidade legal e contratual
- Eficácia das medidas de mitigação de risco

Resultados são documentados e ações corretivas são acompanhadas pelo RSI.

5. APROVAÇÃO

Responsável de Segurança da Informação: Equipa Antixerox
Direção: Direção do BeatRooter
Data: 18/11/2025

Apêndice B - José Couto - 2024026

1. Introdução

Wargaming é muitas vezes descrito como um exercício onde uns tentam atacar sistemas informáticos e outros tentam defender-se. A definição parece correta, mas falha em captar a dimensão estratégica e cognitiva envolvida. A segurança industrial, especialmente quando falamos de PLCs e manipulação de controlo, demonstra que o verdadeiro wargaming exige técnica, antecipação e compreensão profunda do adversário.

Este documento apresenta de forma clara o que é um PLC, como funciona um ataque de manipulação de controlo segundo o MITRE ATT&CK, exemplos reais, impactos, mitigação e a ligação direta com wargaming.

2. O que é um PLC?

Um **PLC (Controlador Lógico Programável)** é o dispositivo que governa processos industriais. Recebe dados de sensores, executa lógica programada e controla atuadores responsáveis por ações físicas como abrir válvulas, ligar bombas ou ajustar motores.

Presente em setores como água, energia, transportes, fábricas e ambientes críticos, o PLC é o ponto onde o digital encontra o mundo físico. Alterações indevidas na sua lógica podem gerar falhas, danos e interrupção operacional.

3. Manipulação de Controlo (MITRE ATT&CK – T0831)

3.1 Enquadramento MITRE

A técnica **T0831 – Manipulation of Control**, da framework **MITRE ATT&CK for ICS**, descreve situações em que um adversário altera parâmetros, lógica, setpoints ou comandos de controlo, com impacto direto no funcionamento físico do processo.

- **ID:** T0831

- **Tática:** Impact
- **Sub-técnicas:** Nenhuma

3.2 Objetivo da técnica

- Modificar o comportamento de um processo industrial.
- Criar falhas físicas ou ambientais.
- Fazer o sistema operar de forma incorreta sem alertar operadores.
- Ou, em casos mais discretos, degradar lentamente equipamentos.

3.3 Como ocorre

- Alteração de lógica ladder ou blocos de função.
- Manipulação de setpoints (temperatura, pressão, nível).
- Injeção de comandos falsos entre PLC e atuadores.
- Modificação de temporizadores ou ramos de decisão.

3.4 Vetores comuns de ataque

- Acesso remoto indevido.
- Redes industriais mal segmentadas.
- Credenciais comprometidas.
- Engenharia social.
- Malware especializado para ICS.

4. Exemplo didático – Concurso de Bolos

Para clarificar a manipulação de controlo num contexto social:

Se dois concorrentes de um concurso de bolos dependem de uma receita escrita e alguém altera discretamente a receita do adversário — mudando quantidades, tempos ou passos — o bolo final será um fracasso, mesmo que ele siga a receita “correta”.

Este cenário traduz exatamente o impacto de alterar lógica ou setpoints num PLC: pequenas alterações geram grandes consequências e o operador nem sempre sabe que algo foi manipulado.

5. Esquema do ataque a um PLC

SENSOR (nível / fluxo)

↓

PLC

→ lógica (ladder / blocos)

↓

Ataque: Manipulation of Control (T0831)

→ alteração de lógica / parâmetros / ramos / temporizadores

↓

ATUADOR (bomba / válvula)

↓

PROCESSO (ex: água)

6. Casos reais de Manipulação de Controlo

6.1 Maroochy Shire, Austrália (2000)

O primeiro incidente conhecido de manipulação industrial sem malware.
Um ex-funcionário adquiriu acesso aos controladores de saneamento e enviou comandos falsos, libertando mais de 800 000 litros de esgoto em espaços públicos. Demonstrou que ataques ICS não precisam de alta complexidade — apenas acesso e intenção.

6.2 Stuxnet (2010)

Malware criado para alterar logicamente PLCs Siemens.
Destruiu centrífugas nucleares através de manipulação de velocidade, enquanto apresentava dados falsos aos operadores.

6.3 Industroyer / Ucrânia 2015

Atacantes enviaram comandos legítimos mas não autorizados para abrir disjuntores elétricos.
O resultado foram apagões que afetaram centenas de milhares de pessoas.

Estes casos provam que o “T0831” é um vetor real, usado por grupos avançados e com impacto mensurável.

7. Consequências potenciais

A manipulação de controlo pode resultar em:

- Danos permanentes em equipamento.
- Interrupção de produção.
- Impacto ambiental grave.
- Perigos à segurança humana.
- Perda financeira e reputacional.
- Encobrimento de atividades maliciosas com dados falsificados.

Apêndice C - Rafael Baptista - 2024134

T1021.002 SMB / Windows admin shares (ATT&CK)

A técnica T1021.002 do MITRE ATT&CK descreve o uso de partilhas de administrador do Windows (como C$, $ADMIN$, IPC$) através do protocolo SMB para realizar movimento lateral dentro de uma rede comprometida.

O atacante utiliza credenciais válidas ou hashes NTLM (Pass-the-Hash) para aceder remotamente a outros sistemas, tendo como objetivo chegar ao domain controller ou obter informações de um utilizador em específico.

O que é SMB/Admin Shares?

O protocolo SMB (Server Message Block) permite acesso remoto a ficheiros, pastas e serviços.
O Windows cria, por defeito, partilhas de admin como:

- **C$** → acesso ao disco C:
- **ADMIN$** → acesso ao diretório Windows
- **IPC$** → canal de comunicação para pipes nomeados

Estas partilhas permitem conexões de admin remotas e também permitem que um atacante, com credenciais válidas, tenha acesso remoto total.

Como funciona o ataque:

1. O atacante obtém credenciais (OSINT, engenharia social, password recycling ou leaks). / Ou obtém hashes NTLM após comprometer a primeira máquina.
2. Liga-se às partilhas via SMB.
3. Faz upload de um payload (ex.: agent.exe) para C$\Windows\Temp ou ADMIN$.
4. Utiliza serviços remotos (PsExec, SMBExec, WMI) para executar o payload.
5. Ganha um shell remoto e continua o movimento lateral.

Porque é que é importante no MITRE?

- Usa funcionalidade legítima do Windows.
- Muitas organizações deixam SMB aberto internamente.
- Fácil escalar privilégios quando se obtém acesso a sistemas, principalmente como admin.
- Pouco ruído na rede.

MITRE D3FEND

D3-FW Network Isolation

- Bloquear tráfego SMB quando as máquinas não precisam.

D3-AC Account Use Policies

- Admin da máquina A não deve funcionar na máquina B.
- Não usar contas administrativas locais repetidas entre máquinas.
- Remover pastas sensíveis escondidas na rede.

Medidas de Deteção

- Alertas para logins no SMB.
- Monitorizar criação de ficheiros em pastas sensíveis.
- Deteção de execução remota via SVCCTL, WinRM ou PsExec.

Modus Operandi

- **Reconhecimento Passivo**
- **Lateral Movement (T1021.002)**

Etapas do Modus Operandi

1. Reconhecimento Passivo

Recolha de informação sem tocar na infraestrutura alvo:

- Pesquisa no LinkedIn, através de engenharia social ou até mesmo encontra leaks online.
- Analisa metadados em ficheiros publicados.
- Pesquisa no Shodan e encontra serviços desatualizados.

2. Seleção do Ponto de Entrada

Identificação de um endpoint fraco que pode ser pivoted:

- PC de um funcionário com SMB aberto.
- Conta com password simples ou leaked.
- Por vezes, até guest accounts.

3. Lateral Movement

1. Aceder às shares administrativas (C, $ADMIN$).
2. Fazer upload do payload para uma das pastas.
3. Execução remota através do PsExec ou SMBExec.
4. Ganha-se acesso pleno ao sistema.

4. Privilege Escalation

Depois da primeira máquina:

- Dump de LSASS para obter credenciais.
- Enumeração do Active Directory.
- Escalation para Domain Admin.

5. Objetivo Final e Impacto

Dependendo da operação:

- Exfiltração de dados.
- Encriptação para ransomware.
- Instalação de backdoors.
- Persistence Techniques.

Apêndice D - Samuel Rocha - 2024127

Relatório de Vulnerabilidade — CWE-489: Active Debug Code

1. Identificação da Vulnerabilidade

- Nome: Active Debug Code

- Código: CWE-489

- Categoria: Erros de Desenvolvimento / Configuração

- Severidade: Média a Alta

2. Descrição Geral

A vulnerabilidade CWE-489 ocorre quando código de debug permanece ativo em produção, expondo informações internas, lógica sensível e comportamentos não documentados que podem comprometer a segurança da aplicação.

3. Porque é uma Vulnerabilidade?

O código de debug pode expor informação sensível, criar comportamentos inesperados e abrir portas para ataques como privilege escalation ou bypass de autenticação.

4. Mapeamento com Frameworks

- MITRE ATT&CK: Discovery, Privilege Escalation, Defense Evasion.

- OWASP: A05 Security Misconfiguration, A07 Authentication Failures, A03 Sensitive Data Exposure.

5. Causas Comuns

- Deploy com DEBUG ativo.

- Endpoints escondidos para testes.

- Logs excessivos.

- Falta de auditoria no código.

6. Impactos

Técnicos: exposição de credenciais, bypass de segurança, fuga de dados.

Organizacionais: perda de confiança, não conformidade com RGPD, riscos legais.

7. Exemplo Concreto

Um endpoint de debug permanece ativo:

```
app.get("/login", (req, res) => {
   req.session.user = "admin";
   res.send("Login efetuado");
});
```

Um atacante descobre a rota, obtém acesso administrativo e compromete o sistema.

8. Mitigações

- Remover totalmente código de debug antes do deploy.

- Code review rigoroso.

- Variáveis de ambiente seguras (DEBUG=false).

- Monitorização ativa pelo Blue Team.

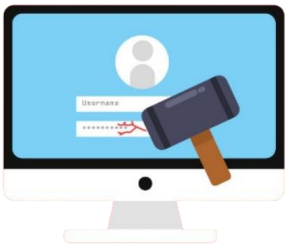
9. Conclusão

A CWE-489 é simples de evitar, mas extremamente perigosa quando negligenciada. Boas práticas DevSecOps e auditorias contínuas são essenciais para prevenir este tipo de falha.

Apêndice E - Davide Ferreira - 2024154

Teste Individual - Davide Ferreira

MITRE ATT&CK - Forced Authentication - Autenticação Forçada



O que é:

É uma técnica usada por atacantes para forçar um sistema ou utilizador a enviar automaticamente as suas credenciais para um servidor controlado pelo atacante.

Como funciona:

O ataque de Autenticação Forçada consiste em levar o sistema da vítima a iniciar automaticamente um processo de autenticação para um servidor controlado pelo atacante. Este processo ocorre sem que o utilizador escreva qualquer senha, pois o sistema operativo.

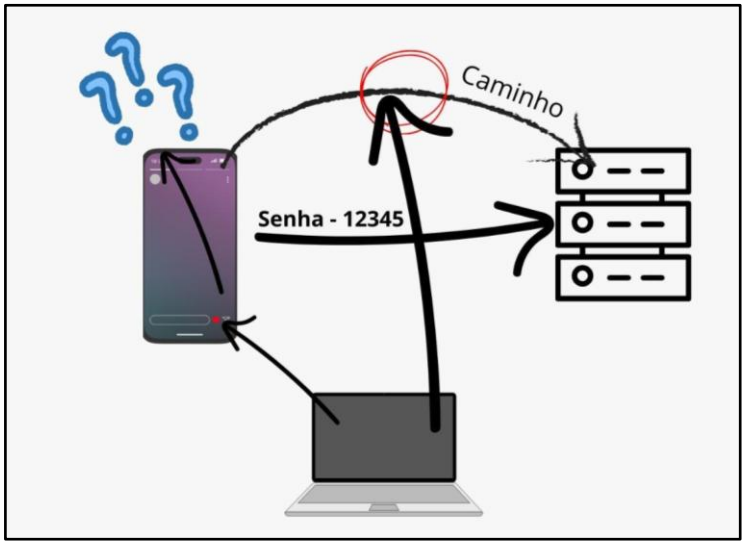
Exemplo de diversas formas estes situações:

Spearphishing: É um documento malicioso que pode conter um recurso que é carregado automaticamente quando o documento é aberto.

Ficheiro LNK ou SCF modificado

Um atalho.LNK(ficheiro de atalho windows) de comandos que pode ser manipulado para apontar o ícone para um caminho externo para obter o recurso, expondo as credenciais das vítimas.

Um Exemplo:



Explicação:

Na minha explicação, temos um dispositivo que vai aceder ao servidor, o qual tem a senha “12345”. Davide Ferreira, usando o seu computador, coloca o telefone sem internet e analisa o caminho de comunicação entre o telefone e o servidor para descobrir a senha e aceder ao servidor como se fosse o telefone.

Webgrafia:

Site- Forced Authentication - Autenticação Forçada

O meu exemplo

Apêndice F - Rodrigo Baltazar - 2024142

Spearphishing Attachment

Como em qualquer ataque de phishing, o Spearphising Attachment conta com a execução da vítima como premissa para o ataque ser realizado, mas com 2 diferenças importantes. A primeira é por se caracterizar como Spearphising, ou seja, o conteúdo do e-mail ou mensagem é personalizado para a vítima, falsificando a mensagem de modo a parecer uma fonte confiável para a vítima. E mais importante, no conteúdo da mensagem há algum ficheiro que ao ser executado pela vítima, o ataque é completado.

Este ataque conta com a engenharia social, pois o objetivo é utilizar a própria vítima como meio de burlar as defesas do sistema que impedia a execução do ficheiro com malware.

Exemplo:

A vitima corriqueiramente recebe e-mail de um colega de trabalho cujo email é jose.machado@gmail.com, o atacante sabendo dessa informação cria um e-mail similar da seguinte forma jose.rnachado@gmail.com com mensagens como:

"Preciso da sua ajuda rápido, o meu computador não quer abrir esse ficheiro, tente abrir no seu e me envie o conteúdo, o pessoal do TI configurou mal o meu e-mail, agora mesmo quando eu dou autorização para o antivírus dizendo que é confiável ele não abre, veja se você consegue, é urgente!!!"

Neste exemplo o e-mail parece mesmo com algo familiar a vítima, e o senso de urgência é sempre elevado, o que torna a vítima ansiosa para resolver a questão. O próprio e-mail também avisa de um possível aviso do antivírus e por parecer de uma fonte confiável, leva a vítima a confiar que trata-se apenas de uma má configuração.

Este tipo de ataque é amplamente utilizado contra alvos considerados importantes, e visam pessoas com acessos privilegiados e cargos de poder que tenham pouco conhecimento em cibersegurança, o que as torna alvos fáceis para os atacantes

Apêndice G - Manifesto

Preâmbulo

Com o objetivo de organizar a mapear ataques em sistemas informáticos adversários em equipa, nós buscamos ilustrar a necessidade de organizar e reportar toda a informação retirada durante a etapa de reconhecimento do sistema alvo, suas tecnologias, modos de operação e vulnerabilidades.

A informação deve estar clara e facilmente acessível a todos os colaboradores do ataque, de modo que todos possam incrementa-la gradualmente, formando um mapa do sistema informático adversário.

A informação deve ser facilmente reconhecida, organizada e visual, de modo que as suas interconexões fiquem claras para todos os atacantes, formando um mapa do *cyber* ataque.

Este mapa deve ser mantido em sigilo e ser transmitido apenas em canal seguro, de modo que esta informação nunca chegue ao adversário.

Princípios

- Armazenar a informação do adversário de forma organizada.
- As informações devem ser interconectadas, quando aplicável.
- Utilizar recursos visuais facilmente compreensíveis.
- Compartilhar toda a informação entre os colaboradores de ataque.
- Notificar os colaboradores do ataque quando alguma informação foi modificada.
- Manter o sigilo, as informações só devem ser compartilhadas entre os colaboradores.
- Manter um canal seguro para troca de informações.
- Coordenar o ataque em equipa, de modo a todos estarem cientes do que está sendo feito.

Ações Práticas

Organização e Estruturação da Informação

1. **Armazenar** todas as informações coletadas em pastas organizadas por tipo, devidamente identificadas quanto à origem.
2. **Relacionar visualmente** as informações por meio de conexões e nós, de modo a criar um diagrama que represente a estrutura e o funcionamento do sistema.
3. **Criar logs** de acesso e modificação das informações, mantendo sempre um backup da versão anterior.

Segurança, Sigilo e Controle de Acesso

4. **Criar um canal seguro** de troca de informação, com acesso exclusivo aos membros autorizados, sendo proibido o compartilhamento externo.
5. **Criptografar** todas as informações coletadas, garantindo que o acesso seja restrito aos membros autorizados.

Comunicação, Atualização e Coordenação da Equipe

6. **Compartilhar** imediatamente com os membros quaisquer novas informações relevantes.
7. **Notificar** todos os membros sempre que alguma informação for atualizada.
8. **Comunicar previamente** aos membros qualquer ação planejada.
9. **Reportar** os resultados obtidos e garantir que sejam revisados posteriormente pelos demais membros.

“War is the realm of uncertainty.” – Carl von Clausewitz

O conhecimento sobre a estrutura e arquitetura do adversário é sempre limitada e incompleta. Estruturar a informação coletada é parte essencial para mapear as vulnerabilidades encontradas e compartilhar entre os atores do ataque. Mais do que apenas dados, é a compreensão e interpretação adequada que dá valor e importância a informação.

Conclusão

O sucesso de um **ciberataque** resulta da combinação de múltiplos fatores técnicos, organizacionais e estratégicos. Mais do que dominar ferramentas e **vulnerabilidades**, é necessário compreender profundamente o funcionamento do **sistema adversário** e selecionar **alvos** capazes de produzir impactos relevantes na sua capacidade operacional. No contexto da **guerra cibernética**, a coordenação entre membros da equipa ofensiva, por meio do compartilhamento de informações e da divisão especializada de tarefas, aumenta a eficácia e acelera o ciclo de ataque.

Ainda assim, nenhum conjunto de procedimentos garante o êxito: cada operação envolve **incertezas** inerentes, características de um ambiente altamente dinâmico e contestado. Por isso, **criatividade**, **adaptação** e **flexibilidade** continuam sendo elementos essenciais para que operações ofensivas possam superar tanto limitações técnicas quanto respostas defensivas inesperadas.