B2-5
アタックサーフェスのその先にあるもの
～リアルな攻撃、リアルな現状～

標的型メール開封後に起きていること

NTTフィールドテクノ（NTT西日本）
四方 直樹

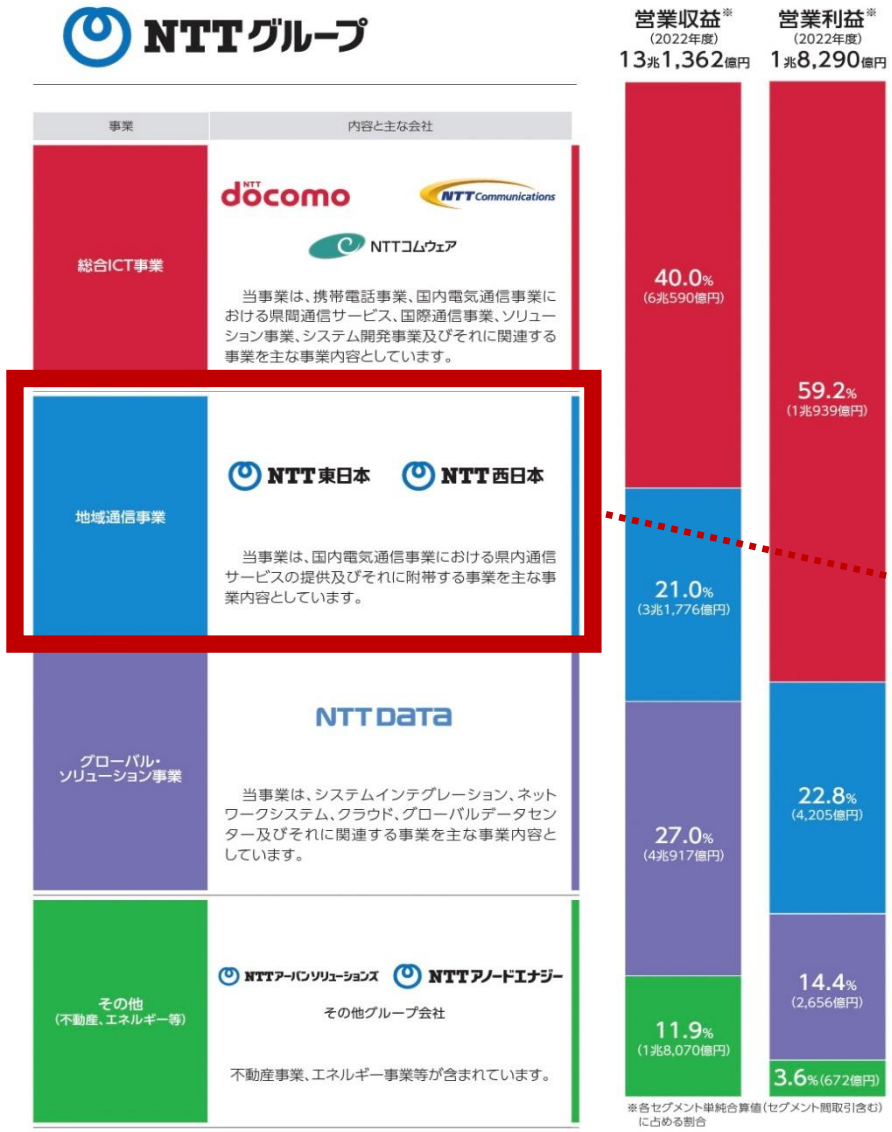
自己紹介

西日本電信電話株式会社（NTT西日本）

四方 直樹 *SHIKATA Naoki*

- 2005年に新卒でNTT西日本に入社
- セキュリティエンジニア、クラウドのプロダクトマネージャや事業戦略立案、海外駐在、米国大学院において情報セキュリティ修士号の取得
- 2022/9よりオフェンシブセキュリティ観点でセキュリティ対策を進める『**レッドチーム**』の立上げに従事

自社紹介



https://group.ntt/jp/ir/library/annual/pdf/integrated_report_23j.pdf

https://www.ntt-west.co.jp/corporate/about/group.html

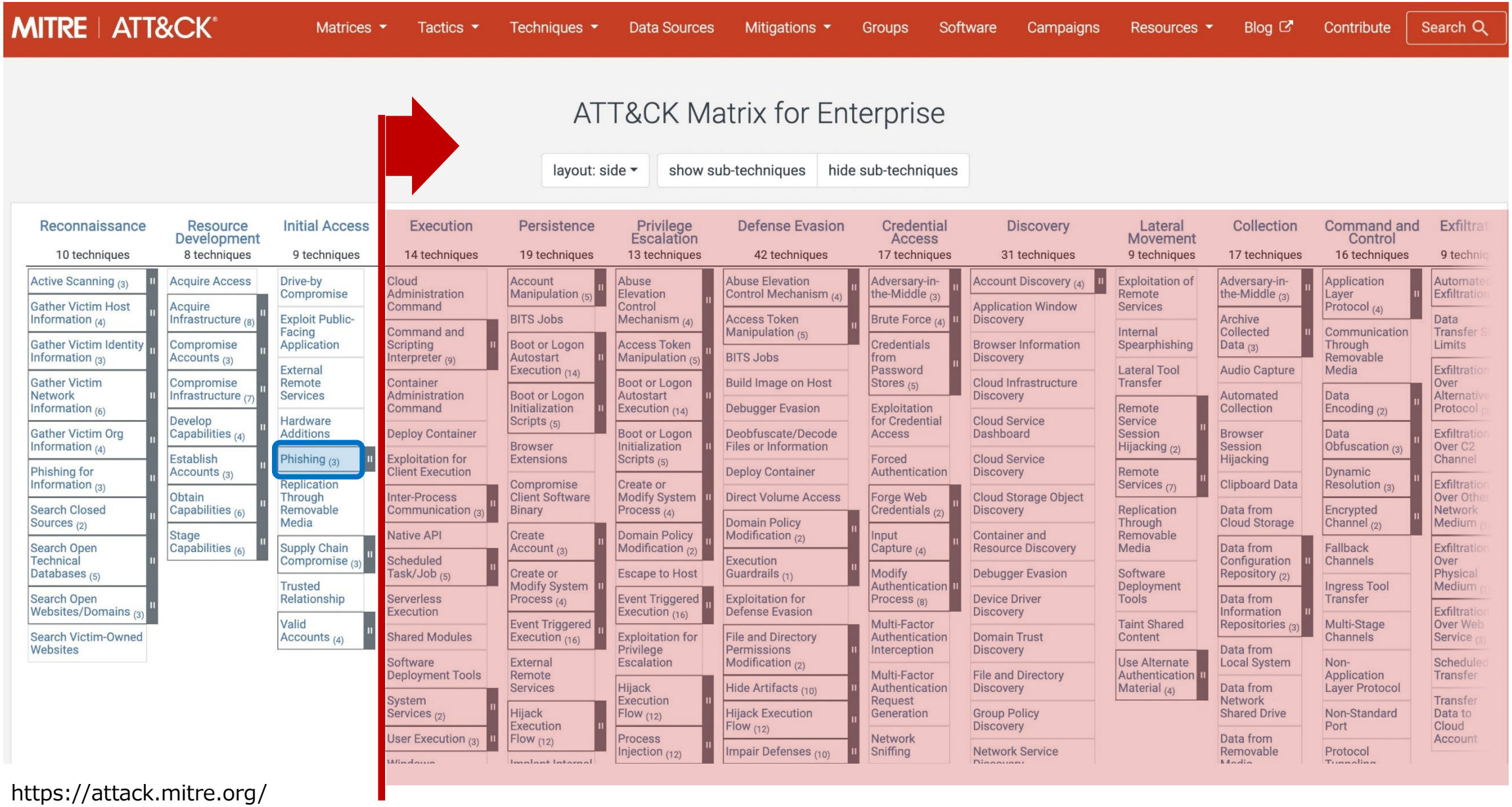
はじめに

標的型メールは開封するな、は認識しているものの
実際に開封したのち、どのようにデータ搾取に繋がっているのか！？

というテーマに対して、（ランサムウェアはともかく）あまり現実感が無く
イメージがつきにくい方もおられるのでは、と思われます。

本日は、攻撃者が内部侵入後に何をするのか、について一部触れたいと思います。

本日のスコープ



https://attack.mitre.org/

共通認識 - 脆弱性をつく/侵入後に内部探索するための情報は世に溢れている -

EXPLOIT DATABASE

Microsoft Windows 11 - 'apds.dll' DLL hijacking (Forced)

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
51733	N/A	MOEIN SHAHABI	LOCAL	WINDOWS	2023-10-09

EDB Verified: × Exploit: / {} Vulnerable App:

```
#----------------
# Title: Microsoft
# Date: 2023-09-01
# Author: Moein Sh
# Vendor: https://
# Version: Windows
# Tested on: Windo
#----------------

Description:

HelpPane object al

Instructions:
```

```
Instructions:

1. Compile dll
2. Copy newly compiled dll "apds.dll" in the "C:\Windows\" directory
3. Launch cmd and Execute the following command to test HelpPane object "[System.Activator]::CreateInstance([Type]::GetTypeFromCLSID('8CEC58AE-07A1-11D9-B15E-000D56BFE6EE'))"
4. Boom DLL Hijacked!


------Code_Poc-------
#pragma once
#include <Windows.h>



// Function executed when the thread starts
extern "C" __declspec(dllexport)
DWORD WINAPI MessageBoxThread(LPVOID lpParam) {
    MessageBox(NULL, L"DLL Hijacked!", L"DLL Hijacked!", NULL);
    return 0;
}

PBYTE AllocateUsableMemory(PBYTE baseAddress, DWORD size, DWORD protection = PAGE_READWRITE) {
#ifdef _WIN64
    PIMAGE_DOS_HEADER dosHeader = (PIMAGE_DOS_HEADER)baseAddress;
    PIMAGE_NT_HEADERS ntHeaders = (PIMAGE_NT_HEADERS)((PBYTE)dosHeader + dosHeader->e_lfanew);
    PIMAGE_OPTIONAL_HEADER optionalHeader = &ntHeaders->OptionalHeader;

    // Create some breathing room
    baseAddress = baseAddress + optionalHeader->SizeOfImage;

    for (PBYTE offset = baseAddress; offset < baseAddress + MAXDWORD; offset += 1024 * 8) {
        PBYTE usuable = (PBYTE)VirtualAlloc(
            offset,
            size,
            MEM_RESERVE | MEM_COMMIT,
            protection);
```

infosecn1nja / AD-Attack-Defense Public

Code | Issues | Pull requests | Actions | Projects | Security | Insights

master | 1 branch | 0 tags | Go to file | Code

infosecn1nja Added BackupOperatorToolkit & Azure hacking resources a22795a on Mar 4 120 commits

README.md Added BackupOperatorToolkit & Azure hacking resources 8 months ago

README.md

Active Directory Kill Chain Attack & Defense

Summary

This document was designed to be a useful, informational asset for those looking to understand the specific tactics, techniques, and procedures (TTPs) attackers are leveraging to compromise active directory and guidance to mitigation, detection, and prevention. And understand Active Directory Kill Chain Attack and Modern Post Exploitation Adversary Tradecraft Activity.

https://www.exploit-db.com/exploits/51733
https://github.com/infosecn1nja/AD-Attack-Defense#defense-evasion

思い浮かぶ疑問

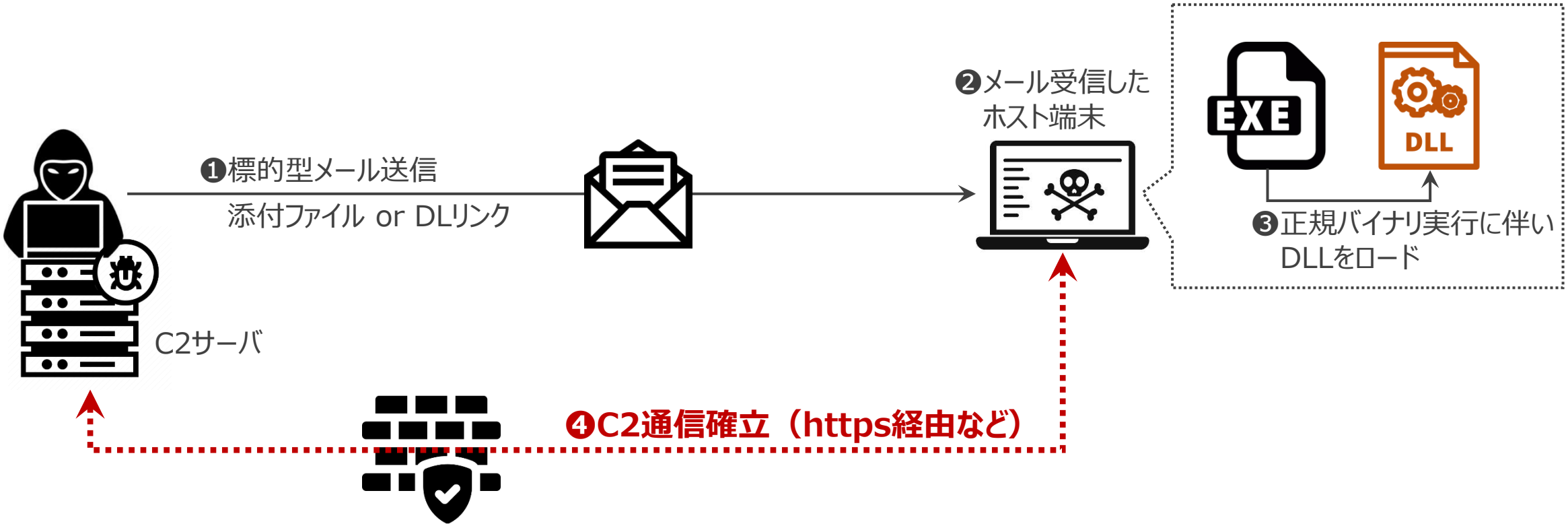
様々な脆弱性に対するexploitコードやハッキングツールが公開されているのは知ってるが、、、

“エンドポイント端末にEPP/EDRなどが導入されている環境でもホントに攻撃は成立するんだっけ？”

攻撃は成立します。
ただ、成功させるためには重要なポイント[※]があります

【※】攻撃者にとっての攻撃侵害コスト

C2サーバとの通信確立 #C2 Establishment

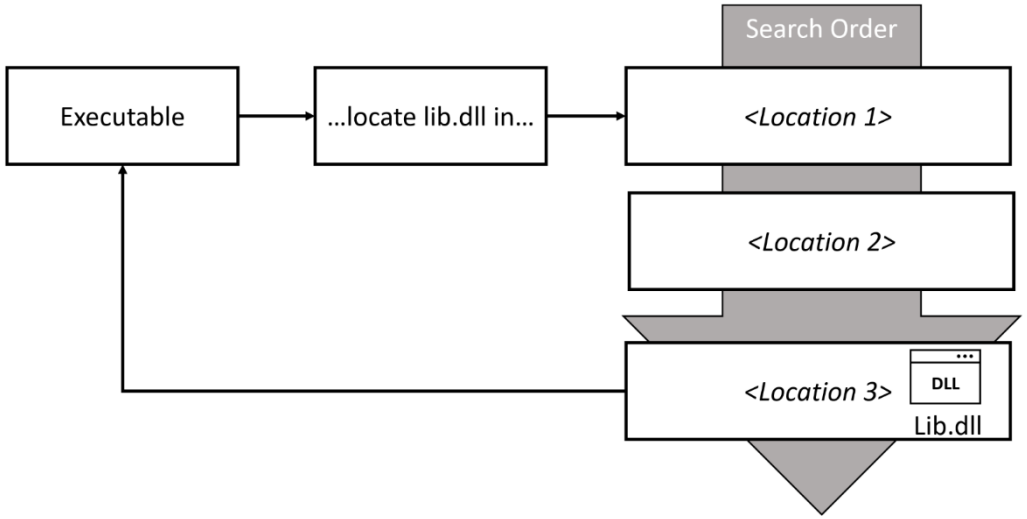


具体的な手法

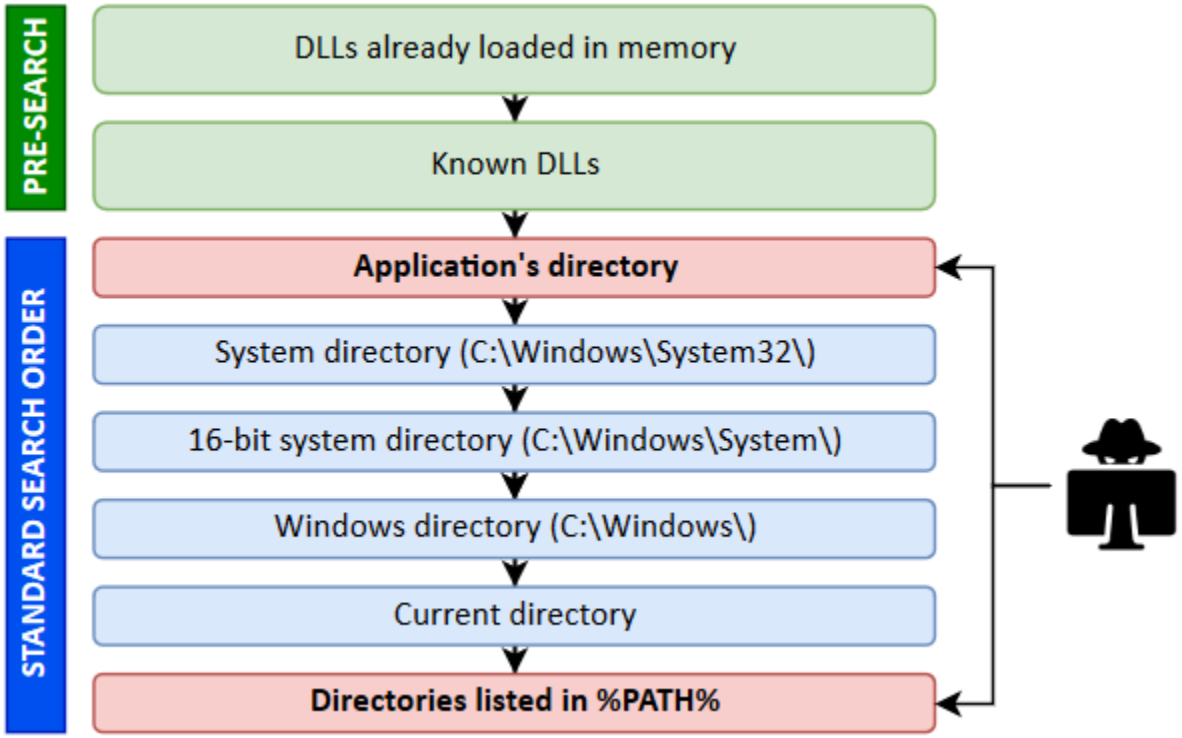
- **正規実行ファイルで悪性DLLをロード**（悪性DLLにC2サーバのbeacon埋め込み）
- **実行ファイル群の偽装**（ショートカット/アイコンの偽装、隠しファイル等）

（参考）DLLサイドローディング #DLL Side-Loading

Mechanism

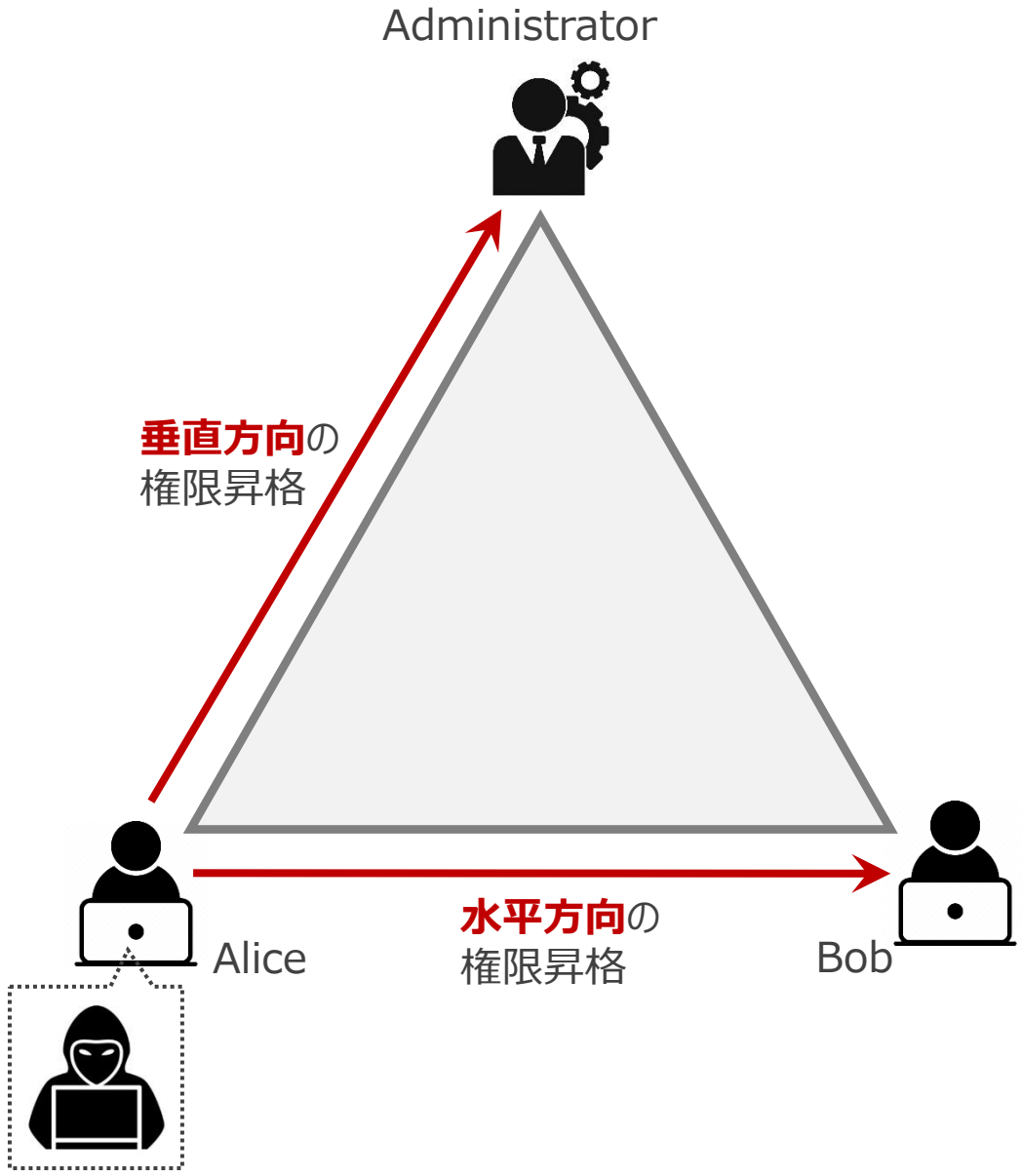


Search Order



https://www.bitdefender.com/blog/businessinsights/tech-explainer-what-is-dll-sideloading/
https://itm4n.github.io/windows-dll-hijacking-clarified/

権限昇格 #Privilege Escalation



水平方向

- **ファイルサーバ等に保管されているCredential情報利用**
  →一般権限であっても各種社内資産にアクセス可能
  になることで、社内システムの権限不備を活用
- **情報の搾取**
  →パケットキャプチャなど

垂直方向

EPP/EDRが入っていると有名なハックツール利用はNGなため
Sticky Keys（sethc.exe等）の利用は難しい

- **UACバイパス**
  →UACバイパス後は、C2フレームワーク内蔵コマンド利用
- **プロセスインジェクション/ハイジャッキング**
  →DLLインジェクション
- **内製アプリケーション**
  →権限設定の不備が散見される

検知回避 #Defense Evasion

具体的な手法

- **難読化**
  →悪性コードの難読化/暗号化、packerの利用
- **正規ファイル利用**
  →OSにBuilt-inされたバイナリを用いて、悪性スクリプトを呼び出す
- **通知の妨害**

https://attack.mitre.org/

さいごに

先述の侵害後も、”水平展開“、”情報搾取“などで様々な手法を
現実の環境でも適用するが、それら含め、攻撃を防ぐためには
（攻撃者にとっての）攻撃侵害コストを高めることが大切


Thank you