Red Team Built-in Scenario Place

16 May 2020
Red Team Village
May'hem Summit

About us

Gokberk Gulgun

Offensive Security Engineer – Red Teamer

Several Certifications

twitter @ggsec_

Erdener Uyan

Application Security Engineer

twitter @sudoeu

Disclaimer: All views expressed in this presentation are our own and do not represent the opinions of our employers

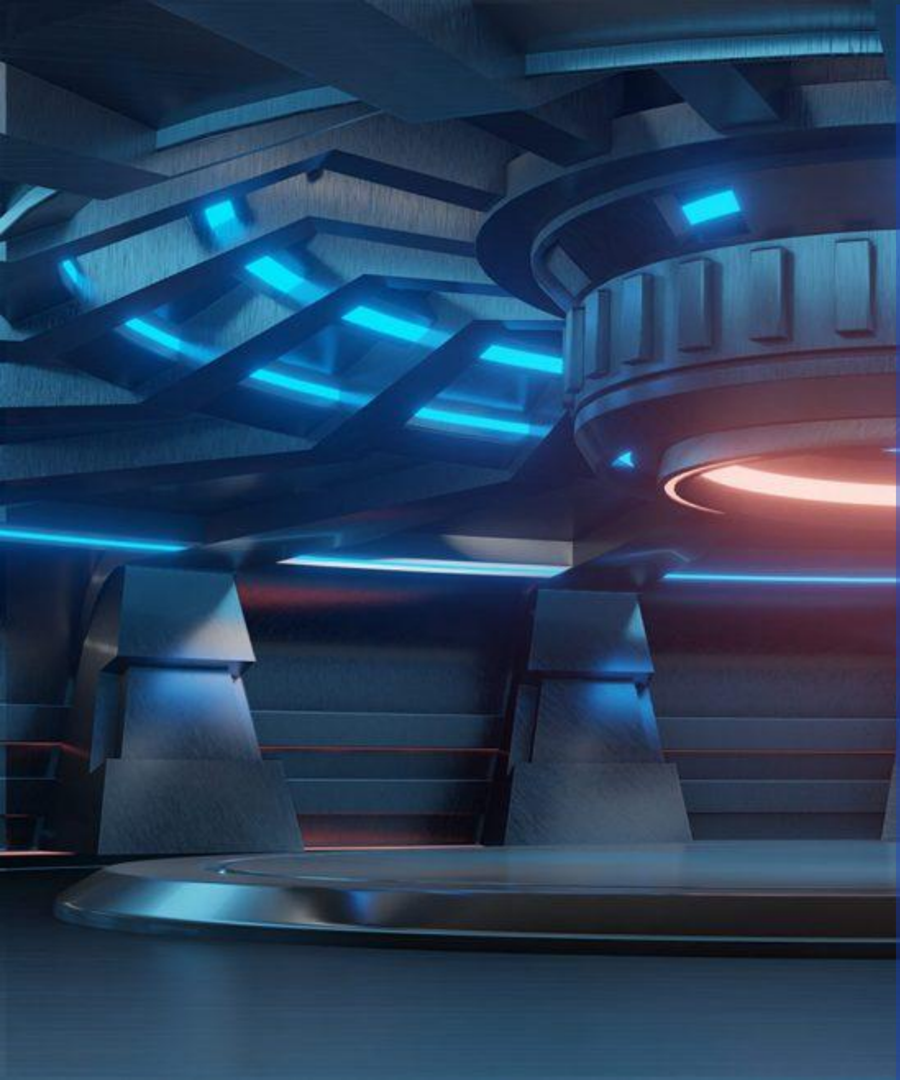
OUTLINE

- Red | Blue | Purple
- Red Team Activities
  - The Cyber Kill Chain
  - Mitre Att&ck Framework
- Challenges
  - Red teams
  - Simulation tools
- Built-in Scenario Place (Manticore)
- Demo
- Future Work

RED TEAM

Offensive Security

- Vulnerability Assessments
- Penetration Tests
- Threat Emulation
  - Social Engineering
  - Physical Security Tests



BLUE TEAM

Defensive Security

- Security Controls
- Security Monitoring
- Incident Response
- Threat Hunting
- Digital Forensics
- Malware Analysis

PURPLE TEAM

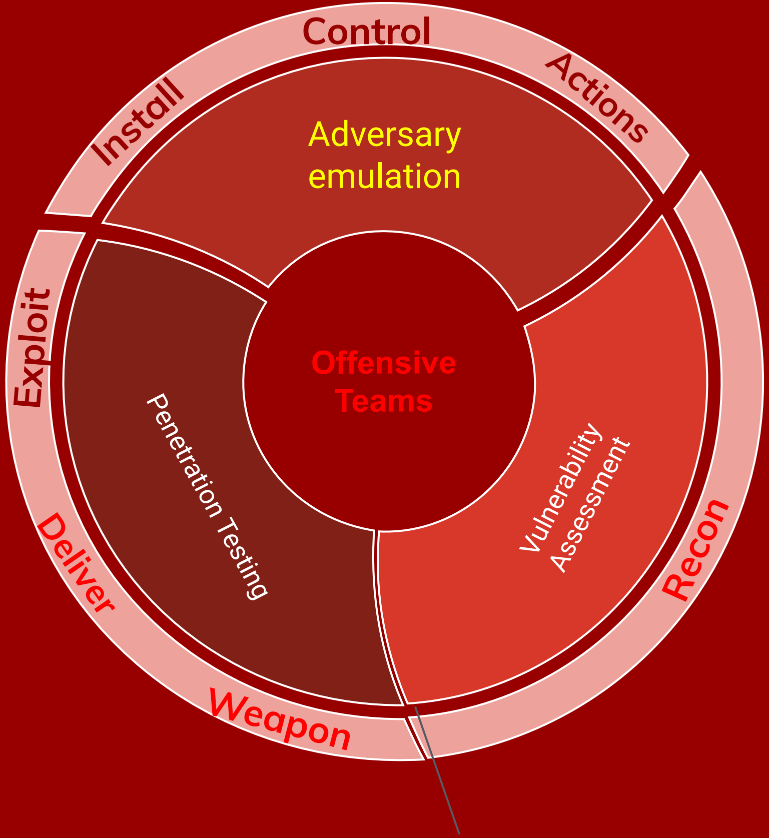
United
Interaction
Iteration
Improvement

ELBONIAN HACKERS ARE TRYING TO STEAL OUR SOURCE CODE.
SEND OUR GOONS TO BEAT THEM UP.
scottadams@aol.com
www.dilbert.com
I WAS THINKING MORE ALONG THE LINES OF IMPROVING OUR DATA SECURITY.
IMPROVE IT OR ELSE I'LL HAVE OUR GOONS BEAT YOU UP.
©2005 Scott Adams, Inc./Dist. by UFS, Inc.
9-12-05
THIS IS SURPRISINGLY MOTIVATIONAL.
YOUSE CALL DAT A FIRE–WALL?

The Cyber Kill Chain

Phases

1. Reconnaissance
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command & Control
7. Actions on objective



Mitre Att&ck Framework

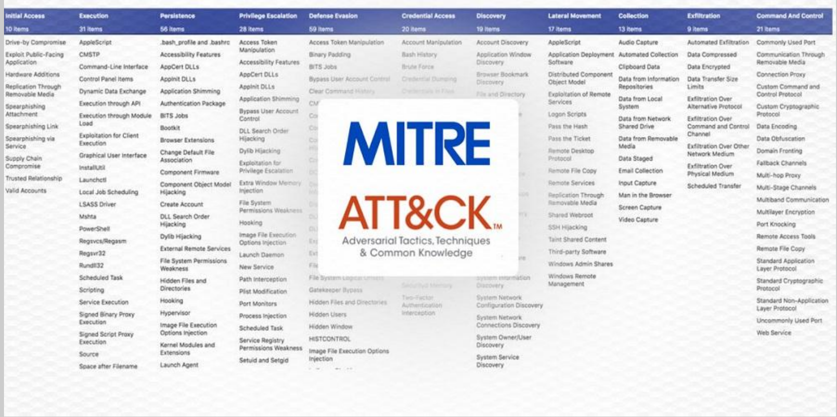
Tactics

1. Initial Access
2. Execution
3. Persistence
4. **Privilege Escalation**
5. Defense Evasion
6. Credential Access
7. Discovery
8. Lateral Movement
9. Collection
10. Command and Control
11. Exfiltration
12. Impact

*https://attack.mitre.org/

Techniques

1. File and Directory Discovery
2. Remote File Copy
3. Registry Run Key/Startup Folder
4. Obfuscated Files or Information
5. Standard Cryptographic Protocol
6. PowerShell
7. **Bypass User Account Control (UAC)**



Initial Access
10 items

- Drive-by Compromise
- Exploit Public-Facing Application
- Hardware Additions
- Replication Through Removable Media
- Spearphishing Attachment
- Spearphishing Link
- Spearphishing via Service
- Supply Chain Compromise
- Trusted Relationship
- Valid Accounts

Execution
31 items

- AppleScript
- CMSTP
- Command-Line Interface
- Control Panel Items
- Dynamic Data Exchange
- Execution through API
- Execution through Module Load
- Exploitation for Client Execution
- Graphical User Interface
- InstallUtil
- Launchctl
- Local Job Scheduling
- LSASS Driver
- Mshta
- PowerShell
- Regsvcs/Regasm
- Regsvr32
- Rundll32
- Scheduled Task
- Scripting
- Service Execution
- Signed Binary Proxy Execution
- Signed Script Proxy Execution
- Source
- Space after Filename

Persistence
56 items

- .bash_profile and .bashrc
- Accessibility Features
- AppCert DLLs
- AppInit DLLs
- Application Shimming
- Authentication Package
- BITS Jobs
- Bootkit
- Browser Extensions
- Change Default File Association
- Component Firmware
- Component Object Model Hijacking
- Create Account
- DLL Search Order Hijacking
- Dylib Hijacking
- External Remote Services
- File System Permissions Weakness
- Hidden Files and Directories
- Hooking
- Hypervisor
- Image File Execution Options Injection
- Kernel Modules and Extensions
- Launch Agent

Privilege Escalation
28 items

- Access Token Manipulation
- Accessibility Features
- AppCert DLLs
- AppInit DLLs
- Application Shimming
- Bypass User Account Control
- DLL Search Order Hijacking
- Dylib Hijacking
- Exploitation for Privilege Escalation
- Extra Window Memory Injection
- File System Permissions Weakness
- Hooking
- Image File Execution Options Injection
- Launch Daemon
- New Service
- Path Interception
- Plist Modification
- Port Monitors
- Process Injection
- Scheduled Task
- Service Registry Permissions Weakness
- Setuid and Setgid

Defense Evasion
59 items

- Access Token Manipulation
- Binary Padding
- BITS Jobs
- Bypass User Account Control
- Clear Command History
- [illegible]
- File System Logical Offsets
- Gatekeeper Bypass
- Hidden Files and Directories
- Hidden Users
- Hidden Window
- HISTCONTROL
- Image File Execution Options Injection

Credential Access
20 items

- Account Manipulation
- Bash History
- Brute Force
- Credential Dumping
- Credentials in Files
- [illegible]
- Securityd Memory
- Two-Factor Authentication Interception

Discovery
19 items

- Account Discovery
- Application Window Discovery
- Browser Bookmark Discovery
- File and Directory Discovery
- [illegible]
- System Information Discovery
- System Network Configuration Discovery
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery

Lateral Movement
17 items

- AppleScript
- Application Deployment Software
- Distributed Component Object Model
- Exploitation of Remote Services
- Logon Scripts
- Pass the Hash
- Pass the Ticket
- Remote Desktop Protocol
- Remote File Copy
- Remote Services
- Replication Through Removable Media
- Shared Webroot
- SSH Hijacking
- Taint Shared Content
- Third-party Software
- Windows Admin Shares
- Windows Remote Management

Collection
13 items

- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Information Repositories
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged
- Email Collection
- Input Capture
- Man in the Browser
- Screen Capture
- Video Capture

Exfiltration
9 items

- Automated Exfiltration
- Data Compressed
- Data Encrypted
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol
- Exfiltration Over Command and Control Channel
- Exfiltration Over Other Network Medium
- Exfiltration Over Physical Medium
- Scheduled Transfer

Command And Control
21 items

- Commonly Used Port
- Communication Through Removable Media
- Connection Proxy
- Custom Command and Control Protocol
- Custom Cryptographic Protocol
- Data Encoding
- Data Obfuscation
- Domain Fronting
- Fallback Channels
- Multi-hop Proxy
- Multi-Stage Channels
- Multiband Communication
- Multilayer Encryption
- Port Knocking
- Remote Access Tools
- Remote File Copy
- Standard Application Layer Protocol
- Standard Cryptographic Protocol
- Standard Non-Application Layer Protocol
- Uncommonly Used Port
- Web Service

CHALLENGES

Red Team - Problems

- Mitigation Management
- Transparency
- Initial Access
- Time Management
- Attack Interfaces
- Instant Changes

Red Team - Adversarial Emulation Problems

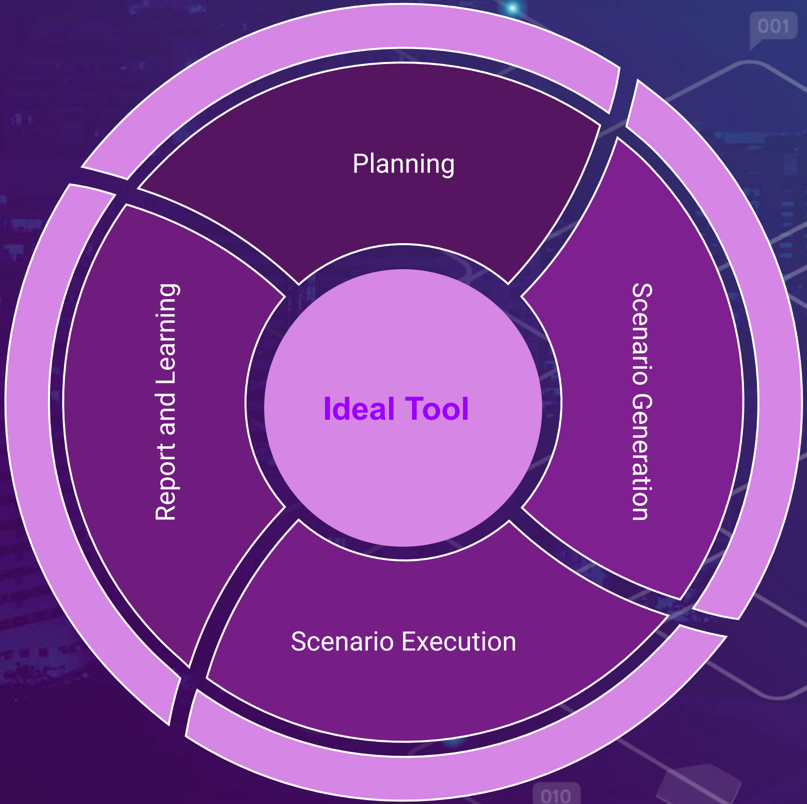
- Cost
- Incident Response
- Tool Availability
- Repeatability
- Unexpected situations

Simulation Tools Problems

- Scenario Transparency
- Scenario Updates
- At most 2000 Scenarios
- Leaving Blue Team dormant
- Open source community
- Complexity
- Reporting

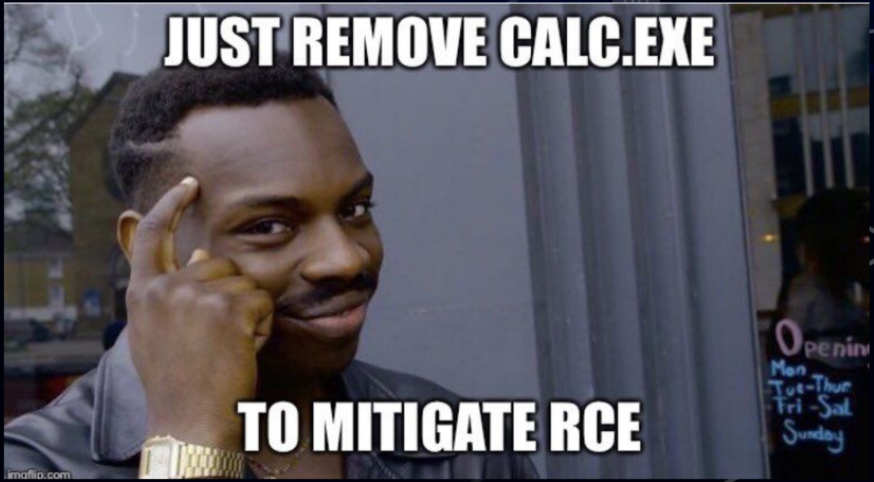
Great Expectations

- Make Emulations Real
- End-to-End
- Repeatable

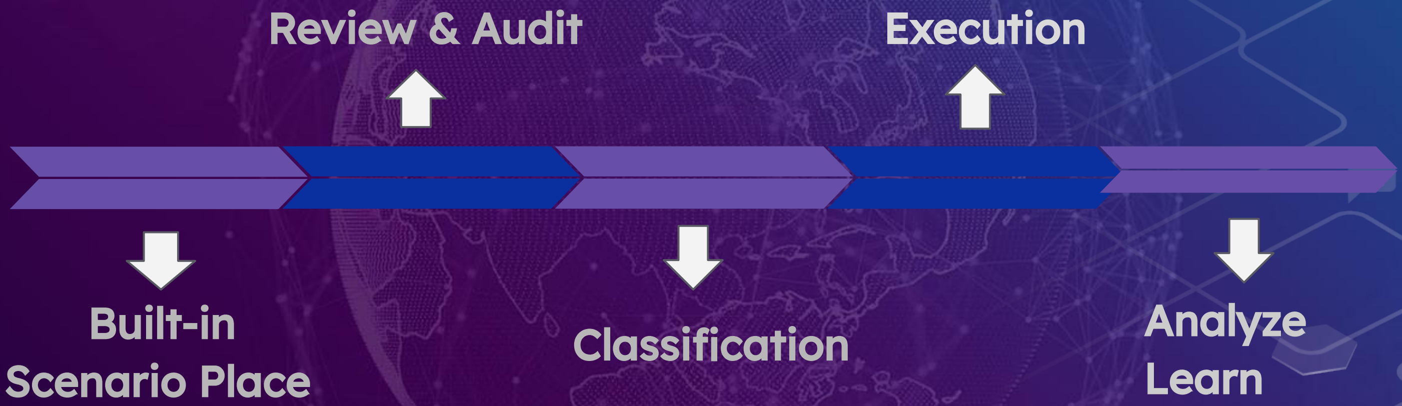


Built-in Scenario Place

- Less complicated
- Make it transparent for red and blue teams.
- Blue Team detection and prevention capabilities
- Distribution of scenarios (Atomic Red Team)
- Recreate APT Scenarios
- Network Based Attacks



How to utilize Built-in Scenario Place



Threat Intelligence Service

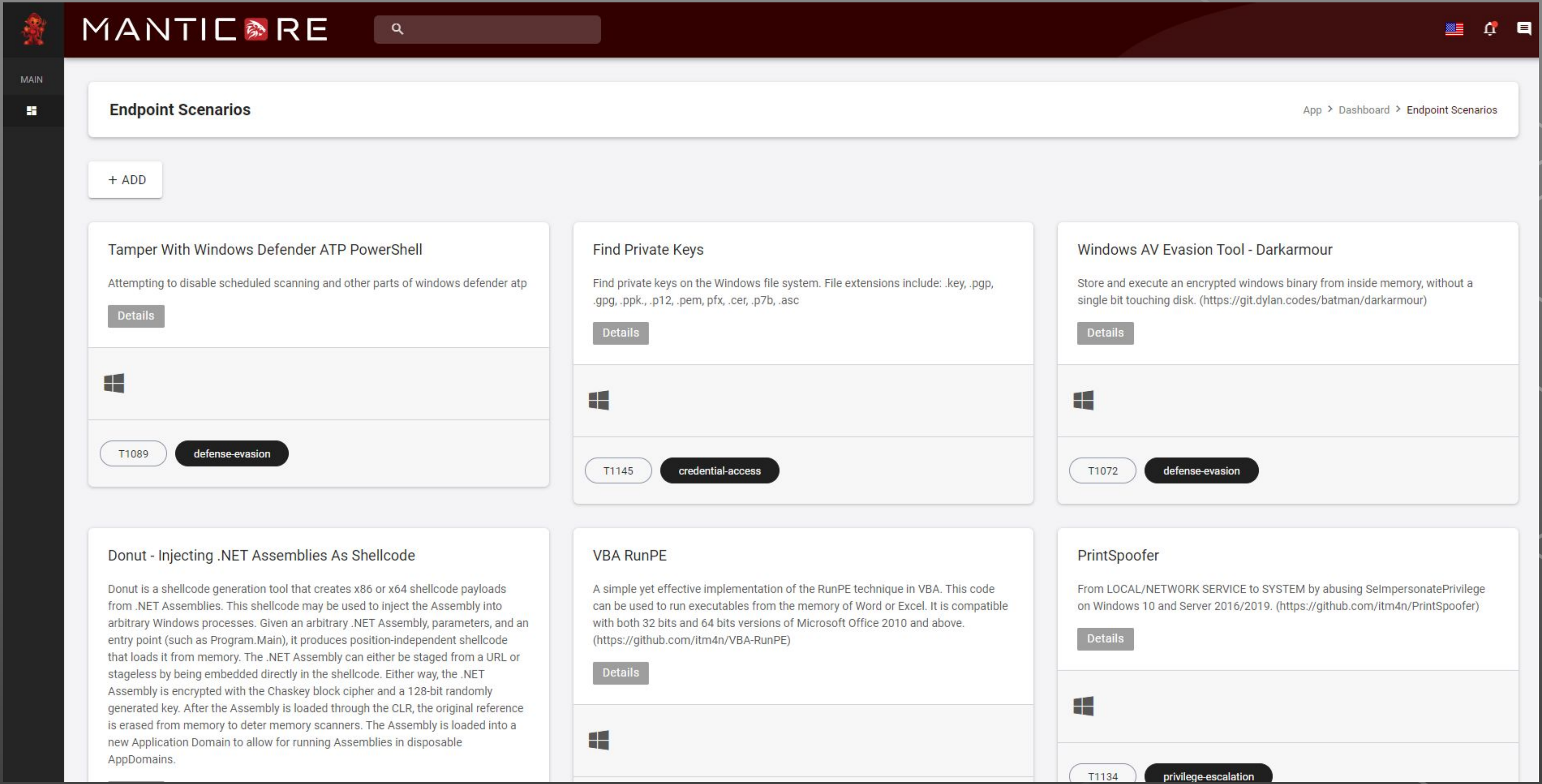
- Automatically recreate APT attack scenarios
- Open source red-team tools as an input
- IOC based network packet generation
- APT Group attack based categorization
- AI based attack generation

About Demonstration

- Network based Scenario Section
- Endpoint based Scenario Section
- APT Groups Section
- Blue Team Techniques and Tactics

DEMO

DEMO



Endpoint Scenarios

Tamper With Windows Defender ATP PowerShell

Attempting to disable scheduled scanning and other parts of windows defender atp

Find Private Keys

Find private keys on the Windows file system. File extensions include: .key, .pgp, .gpg, .ppk., .p12, .pem, pfx, .cer, .p7b, .asc

Windows AV Evasion Tool - Darkarmour

Store and execute an encrypted windows binary from inside memory, without a single bit touching disk. (https://git.dylan.codes/batman/darkarmour)

Donut - Injecting .NET Assemblies As Shellcode

Donut is a shellcode generation tool that creates x86 or x64 shellcode payloads from .NET Assemblies. This shellcode may be used to inject the Assembly into arbitrary Windows processes. Given an arbitrary .NET Assembly, parameters, and an entry point (such as Program.Main), it produces position-independent shellcode that loads it from memory. The .NET Assembly can either be staged from a URL or stageless by being embedded directly in the shellcode. Either way, the .NET Assembly is encrypted with the Chaskey block cipher and a 128-bit randomly generated key. After the Assembly is loaded through the CLR, the original reference is erased from memory to deter memory scanners. The Assembly is loaded into a new Application Domain to allow for running Assemblies in disposable AppDomains.

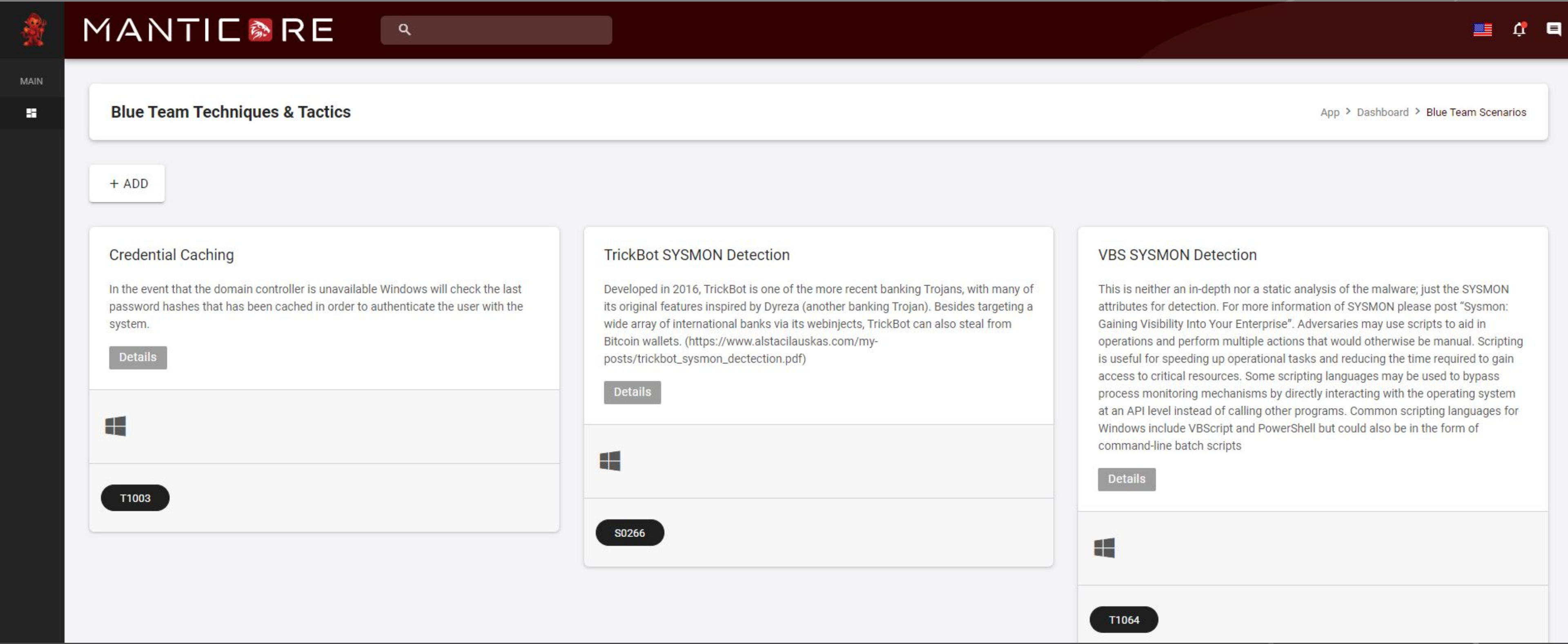
VBA RunPE

A simple yet effective implementation of the RunPE technique in VBA. This code can be used to run executables from the memory of Word or Excel. It is compatible with both 32 bits and 64 bits versions of Microsoft Office 2010 and above. (https://github.com/itm4n/VBA-RunPE)

PrintSpoofer

From LOCAL/NETWORK SERVICE to SYSTEM by abusing SeImpersonatePrivilege on Windows 10 and Server 2016/2019. (https://github.com/itm4n/PrintSpoofer)

DEMO



Blue Team Techniques & Tactics

Credential Caching

In the event that the domain controller is unavailable Windows will check the last password hashes that has been cached in order to authenticate the user with the system.

Details

T1003

TrickBot SYSMON Detection

Developed in 2016, TrickBot is one of the more recent banking Trojans, with many of its original features inspired by Dyreza (another banking Trojan). Besides targeting a wide array of international banks via its webinjects, TrickBot can also steal from Bitcoin wallets. (https://www.alstacilauskas.com/my-posts/trickbot_sysmon_dectection.pdf)

Details

S0266

VBS SYSMON Detection

This is neither an in-depth nor a static analysis of the malware; just the SYSMON attributes for detection. For more information of SYSMON please post "Sysmon: Gaining Visibility Into Your Enterprise". Adversaries may use scripts to aid in operations and perform multiple actions that would otherwise be manual. Scripting is useful for speeding up operational tasks and reducing the time required to gain access to critical resources. Some scripting languages may be used to bypass process monitoring mechanisms by directly interacting with the operating system at an API level instead of calling other programs. Common scripting languages for Windows include VBScript and PowerShell but could also be in the form of command-line batch scripts

Details

T1064

Future

* Increase the number of scenarios

* Make scenarios executable

* Add scheduled scenarios feature

* Integrate a threat intelligence service

* Integrate ML based scenario generation

Thanks!

Any questions?

Please Join the #red-team-talks Channel

https://redteamvillage.io/discord

Waiting for your support!

You can find us at:

info@manticore.zone

https://github.com/Manticore-Platform