HERE TO STAY:
GAINING PERSISTENCE
BY ABUSING ADVANCED
AUTHENTICATION MECHANISMS

MARINA SIMAKOV
IGAL GOFMAN

DEF CON
ABOUT US

Marina Simakov
Security Researcher @ Microsoft

- Holds an M.Sc. in computer science
- Special interest in graph theory
- Huge dog lover

Igal Gofman
Security Researcher @ Microsoft

- Self taught security researcher & developer
- Loves Python
- Electronic music geek!
WHY ARE WE HERE?

- AWARENESS TO WEAK SPOTS
  - Relatively Easy to Exploit
  - Easily Automated
  - Usually not Monitored

- DETECTION METHODS

- TALK TO US
AGENDA

- INTRODUCTION TO KEY TERMS
  - Kerberos
  - Kerberos Delegation

- GAINING PERSISTENCE
  - Focus on High Privileges
  - Malicious JIT Administration

- DEMO

- MITIGATIONS & TAKEAWAYS
INTRODUCTION

- KERBEROS INTRO
  - Ticket based authentication protocol

Diagram:
- Domain User
- LSASS (KERBEROS)
- KDC
- AS
- TGS
- TGT
- AS-REQ {Service:krbtgt}
- AS-REP
- TGS-REQ {Service:app}
- TGS-REP
- ST
- AP-REQ
- App Server

< HERE TO STAY: GAINING PERSISTENCE BY ABUSING ADVANCED AUTHENTICATION MECHANISMS @ DEF CON >
Why?
- An application reusing user credentials
- Web server accessing a SQL DB

How?
- Request tickets on behalf of the user
INTRODUCTION

- DELEGATION TAB
  - UNCONSTRAINED DELEGATION
    • Delegation to any service
  - CONSTRAINED DELEGATION
    • Kerberos Only (S4U2Proxy)
    • Protocol Transition (S4U2Self + S4U2Proxy)
FROM ZERO TO HERO

No Access

Domain Account or Local System Access

Pivoting

Pass-The-Hash
Pass-The-Ticket
BloodHound
GoFetch

Privileged

Pass
-
The
-
Hash

The
-
Ticket

BloodHound

GoFetch

Busted!

Acls

Constrained Delegation

GPO Tampering

Persistence

Domain Admin

Enterprise Admin

HERE TO STAY: GAINING PERSISTENCE BY ABUSING ADVANCED AUTHENTICATION MECHANISMS @ DEF CON
PERSISTENCE — PRIVILEGED

< HERE TO STAY: GAINING PERSISTENCE BY ABUSING ADVANCED AUTHENTICATION MECHANISMS >
# PERSISTENCE - PRIVILEGED

<table>
<thead>
<tr>
<th>COMMON METHODS</th>
<th>WEAKNESSES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dump NTDS.dit (VSS, DRSUAPI)</td>
<td>Detect replication requests from a non-DC machine</td>
</tr>
<tr>
<td>Golden ticket</td>
<td>Detect crafted tickets</td>
</tr>
<tr>
<td>Skeleton key (or others backdoors)</td>
<td>Encryption downgrade</td>
</tr>
</tbody>
</table>
MALICIOUS
JUST-IN-TIME ADMINISTRATION
(MAL-JIT)
PERSISTENCE - PRIVILEGED

- JIT [JUST-IN-TIME] ADMINISTRATION
  - Accounts holding permanent high privileges serve as valuable targets for attackers
  - JIT Administration
    - High privileges are required to perform an operation
    - Get the required privileges for a limited amount of time
    - When the time period expires, the high privileges are revoked
  - Reduces the attack surface
PERSISTENCE - PRIVILEGED

- **MAL-JIT (MALICIOUS-JUST-IN-TIME)**
  - Get administrative access for a limited time
  - Perform malicious operations
  - Leave no traces behind to avoid detection

- **SCENARIOS**
  - Delegation scenario
  - AdminSDHolder scenario 1
  - AdminSDHolder scenario 2
MALICIOUS JIT DELEGATION SCENARIO
**PERSISTENCE - PRIVILEGED**

- **OBJECTS INTRO**
  - Discretionary Access Control List (DACL)
  - Access control entry (ACE)
  - Object Ownership

LSA Matches SIDs from the access Token with SIDs in the ACEs

**Access Token**
- User SID
- Security Group SIDs
- User Rights

**DACL**
- Deny ACEs
- Allow ACEs
- Inherited Deny ACEs
- Inherited Allow ACEs

*HERE TO STAY* GAINING PERSISTENCE BY ABUSING ADVANCED AUTHENTICATION MECHANISMS
PERSISTENCE - PRIVILEGED

- DELEGATION SCENARIO
  - Add a new machine account
  - Set machine owner to a malicious account
  - Owner can edit the ACL of the object
PERSISTENCE - PRIVILEGED

- DELEGATION SCENARIO
  - GPO Tampering: “Enable accounts to be trusted for delegation”
I lost all of my administrative rights
PERSISTENCE - PRIVILEGED

- DELEGATION SCENARIO – MAL-JIT
  - DACL Modification:
    - Add ‘GenericAll’ ACE for malicious_user on new machine
PERSISTENCE - PRIVILEGED

- DELEGATION SCENARIO - MAL-JIT
  - Allow delegation to ‘krbtgt’
  - Request ‘Administrator’ TGT
  - Remove footprints:
    - Remove delegation
    - Revert ACL
  - Perform malicious operations
MAL-JIT SUCCESSFUL!
MALICIOUS JIT
ADMINSDHOLDER MANIPULATION 1
• Persistence can be obtained by ACL modification of privileged groups – such as ‘Domain Admins’

• Problem: AdminSDHolder!
PERSISTENCE - PRIVILEGED

ADMINSDHOLDER MANIPULATION 1

- SDProp overrides the ACLs of protected groups & users with the AdminSDHolder ACL
- Runs periodically (default: 1 hour)
- Result: malicious_user loses his permission

- Protected accounts:
  - Enterprise Admins
  - Domain Admins
  - Administrators
  - ...

< HERE TO STAY: GAINING PERSISTENCE BY ABUSING ADVANCED AUTHENTICATION MECHANISMS @ DEFCON >
The following groups can be excluded from the AdminSDHolder protection:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Group to Exclude</th>
<th>Binary Value</th>
<th>Hexadecimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Account Operators</td>
<td>0001</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>Server Operators</td>
<td>0010</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Print Operators</td>
<td>0100</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Backup Operators</td>
<td>1000</td>
<td>8</td>
</tr>
</tbody>
</table>

Exclude groups by editing the ‘dsHeuristics’ attribute under the Configuration Container.
PERSISTENCE - PRIVILEGED

**ADMINSDHOLDER MANIPULATION 1**

- Add ACEs to excluded groups

- SDProp will not affect the new ACL
- Malicious JIT at any time!
MALICIOUS JIT
ADMINSDHOLDER MANIPULATION 2
PERSISTENCE - PRIVILEGED

- **ADMINSDHOLDER MANIPULATION 2**
  - Change the owner of the AdminSDHolder object
  - Still not allowed to modify group memberships
Oh, No... Not Again!
PERSISTENCE - PRIVILEGED

- **AdminSDHolder Manipulation 2 - Mal-JIT**
  - Just before SDProp is scheduled to run
  - Add malicious_user to the AdminSDHolder ACL
PERSISTENCE - PRIVILEGED

- **ADMINS Rhoulder MANIPULATION 2 - MAL-JIT**
  - SDProp adds malicious ACE to protected objects
Add malicious_user to ‘Domain Admins’ group

Obtain administrative TGT

Revert (erase footprints)
  • Remove malicious_user from AdminSDHolder’s ACL
  • Force SDProp to run: removes malicious ACE from protected groups
  • Remove malicious_user from ‘Domain Admins’ group

Entire operation can be automated to run in seconds!
> **MITIGATIONS**

- **DETECT DELEGATION MISCONFIGURATIONS**
  - GPO: ‘Enable accounts to be trusted for delegation’
  - Monitor accounts trusted for delegation
  - ‘AllowedToDelegateTo’ attribute
  - ‘Account is sensitive and cannot be delegated’

- **MONITOR ADMINS/DHOLDER**
  - ACL
  - Owner
  - Excluded groups

- **EVENT LOGGING**
QUESTIONS ?