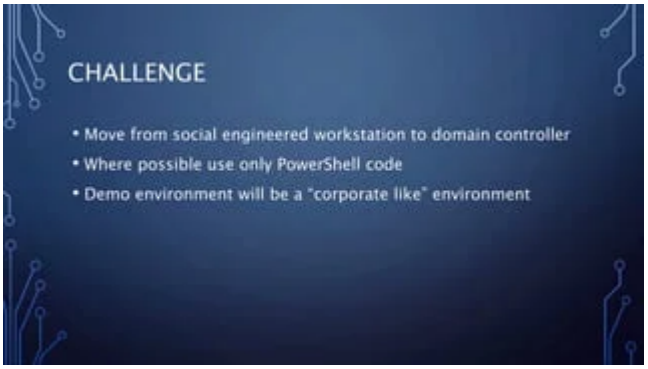
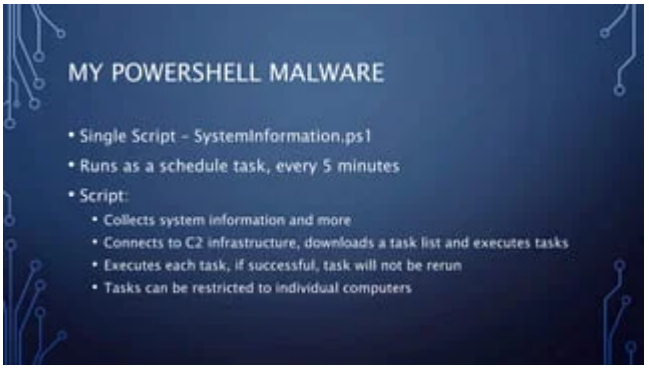
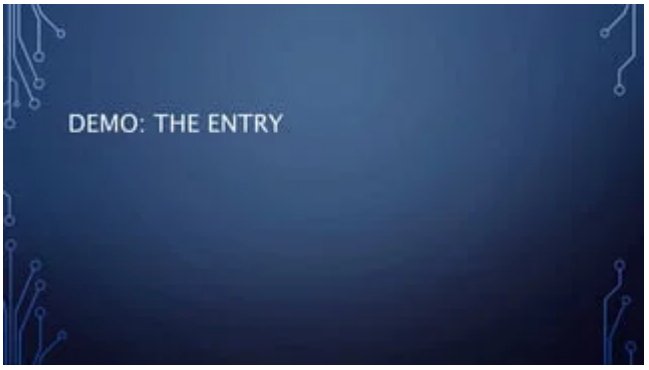
Lateral Movement with PowerShell



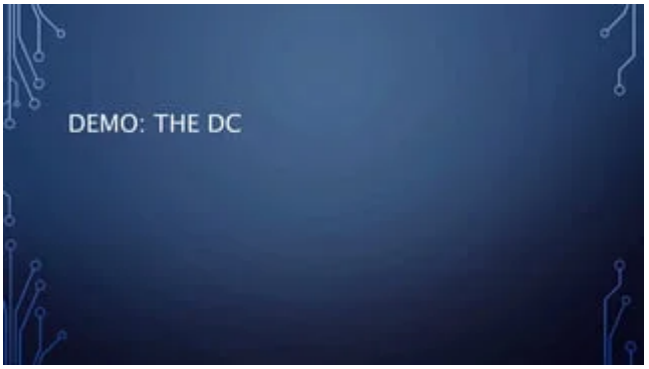

ADVANTAGES AS AN ATTACK PLATFORM
• Code is very easy to develop
• Windows integration
• Plenty of remote execution options
• Designed for automation against 1 - 10000000 devices
• Limited security model
• Antivirus products are no real concern/limitation
• Scripts can be easily hidden from administrators
• Installed by DEFAULT


REAL WORLD POWERSHELL MALWARE
• Prior to March 2014, only a few minor instances
• PowerWorm:
• Infect's Word and Excel documents, initial infection via macro in .doc/.xls
• First spotted by TrendMicro, analysis and rewrite by Matt Graeber (@Mattifestation)
• PoshKoder/PoshCoder:
• PowerWorm crossed with CryptoLocker
• Bitcoin ransom

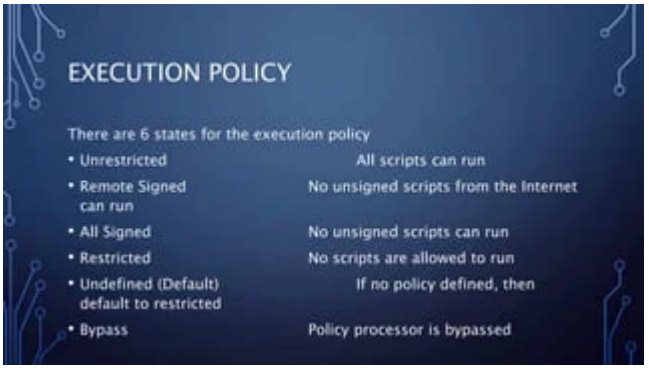

MY POWERSHELL MALWARE
• Single Script - SystemInformation.ps1
• Runs as a schedule task, every 5 minutes
• Script:
• Collects system information and more
• Connects to C2 infrastructure, downloads a task list and executes tasks
• Executes each task, if successful, task will not be rerun
• Tasks can be restricted to individual computers


DEMO: THE ENTRY


WINDOWS POWERSHELL REMOTING AND WINRM
• PowerShell Remoting is based upon WinRM, Microsoft's WS-Management implementation
• Supports execution in 3 ways:
• Remote enabled commands
• Remotely executed script blocks
• Remote sessions
• Security Model = Trusted Devices + User Credentials
• WinRM is required for the Windows Server Manager
• WinRM is enabled by DEFAULT on Windows 2012(R2) Server
• WinRM is allowed through Windows Firewall on all network profiles!


DEMO: THE DC


POWERSHELL SECURITY FEATURES
• Administrative rights
• UAC
• Code Signing
• Local or Remote source using zone.identifier alternate data stream
• PowerShell Execution Policy


EXECUTION POLICY
There are 6 states for the execution policy
• Unrestricted All scripts can run
• Remote Signed No unsigned scripts from the Internet
can run
• All Signed No unsigned scripts can run
• Restricted No scripts are allowed to run
• Undefined (Default) If no policy defined, then
default to restricted
• Bypass Policy processor is bypassed

BYPASSING EXECUTION POLICY

- Simply ask PowerShell: *powershell.exe -executionpolicy unrestricted*
- Switch the files zone.idenfier back to local: *unblock-file yourscript.ps1*
- Read the script in and then execute it (may fail depending on script)
- Encode the script and use -encodedcommand → always works!!!!!
- Get/Steal a certificate, sign script, run script

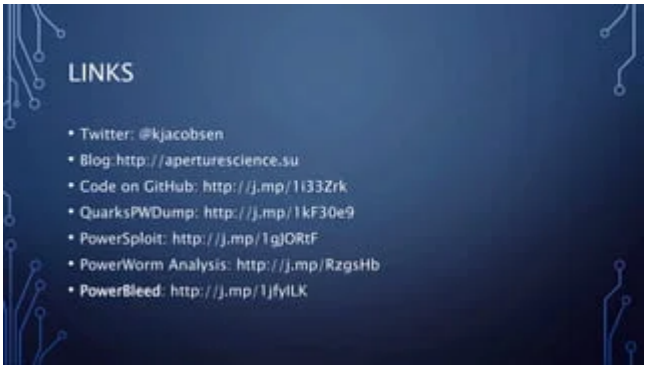
DEMO: THE HASHES

DEFENCE OF THE DARK ARTS

- Restricted/Constrained Endpoints
- Change WinRM Listener
- Change Windows Firewall settings
- Turn it off WinRM
- Application whitelisting

WINRM, NOT JUST AN INTERNAL ISSUE

By default, Microsoft Azure virtual machines expose HTTPS listener to the Inter



More Related Content

PDF

Silent web app testing by example - BerlinSides 2011

PDF

Meraki Solution Overview

PPT

LDAP

PPTX

Introduction to Firebase

DOC

4+ yrs_Exp .Net Resume

PDF

VMworld 2014: vSphere Distributed Switch

PPTX

Citrix XenDesktop and XenApp 7.5 Architecture Deployment

PDF

Segurança na Nuvem

What's hot

DOC

Yasmin-Manual plus Mainframe Testing

PPTX

SCCM_Overview_Updated.pptx

PPTX

Infraestructura como codigo

PPTX

Introduction to Azure AD and Azure AD B2C

DOC

.net Developer Resume(Ming Zhao)

PPTX

Dynamics CRM 2011 Architecture Overview

PDF

Intorduction to Datapower

PDF

TechnicalTerraformLandingZones121120229238.pdf

PPTX

DevSecOps: Key Controls to Modern Security Success

PPTX

AWS Cloud Security

PPTX

An Introduction to OAuth2

PDF

ELK in Security Analytics

PDF

HTTP Request Smuggling via higher HTTP versions

PDF

Access Security - Privileged Identity Management

PPTX

REST API in Salesforce

PPTX

Introduction to appDynamics

PDF

Introduction to Kong API Gateway

PDF

End User Monitoring with AppDynamics - AppSphere16

PPTX

Kafka Retry and DLQ

ODP

OAuth2 - Introduction

Viewers also liked

PPTX

Lateral Movement with PowerShell

PPTX

Exploiting MS15-034 In PowerShell

PDF

The Dark Side of PowerShell by George Dobrea

PPTX

Advanced PowerShell Automation

PPTX

Evolving your automation with hybrid workers

PPTX

DevSecOps in 10 minutes

PPTX

Chef Hack Day Denver

PPTX

Lateral Movement by Default

PDF

Deception Driven Defense - Infragard 2016

PDF

Ansible for the Impatient Devops

PDF

Puppetconf2016 Puppet on Windows

PDF

Lateral Movement: How attackers quietly traverse your Network

PPTX

Building Windows Images with Packer

PPTX

Compliance as Code: Velocity with Security - Fraser Pollock, Chef

PPTX

Lateral Movement - Phreaknik 2016

PPTX

Fun with the Hak5 Rubber Ducky

PPTX

Enabling Enterprise Mobility

PDF

Advanced Threats and Lateral Movement Detection

PPTX

Global Azure Bootcamp 2016 - Azure Automation Invades Your Data Centre

PPTX

DirectAccess, do’s and don’ts

Similar to Lateral Movement with PowerShell

PDF

Powering up on PowerShell - BSides Charleston - Nov 2018

PPTX

Powering up on power shell avengercon - 2018

PPTX

Powering up on PowerShell - BSides Greenville 2019

PDF

Who Should Use Powershell? You Should Use Powershell!

PPTX

Building an Empire with PowerShell

PPTX

Drilling deeper with Veil's PowerTools

PPTX

Enterprise PowerShell for Remote Security Assessments

PDF

PowerShell Defcon for Cybersecurity Topics

PPTX

Client side attacks using PowerShell

PDF

Empire Work shop

PPT

PowerShell Remoting

PDF

DEF CON 23 - Rich Kelley - harness powershell weaponization made easy

PPTX

Pwning the Enterprise With PowerShell

PPTX

DFIR Austin Training (Feb 2020): Remote Access & Deploying Agents

PPTX

Incorporating PowerShell into your Arsenal with PS>Attack

PPTX

Harness: PowerShell Weaponization Made Easy (or at least easier)

PPTX

Kheirkhabarov24052017_phdays7

PPTX

Горизонтальные перемещения в инфраструктуре Windows

PDF

Windows PowerShell Remoting Presentation.pdf

PPTX

Managing enterprise with PowerShell remoting

More from kieranjacobsen

PPTX

The Boring Security Talk - Azure Global Bootcamp Melbourne 2019

PPTX

CrikeyCon VI - The Boring Security Talk

PPTX

The Boring Security Talk

PPTX

The Boring Security Talk

PPTX

Secure Azure Deployment Patterns

PPTX

Ransomware 0, Admins 1

PPTX

Ransomware 0 admins 1

PPTX

DecSecOps in 10 minutes

PPTX

Infrastructure Saturday - Level Up to DevSecOps

PPTX

Dev Breakfast: Level up to DevSecOps

PPTX

DevSecOps - CrikeyCon 2017

PPTX

Azure automation invades your data centre

PPTX

Infrastructure Saturday 2011 - Understanding PKI and Certificate Services

Lateral Movement with PowerShell

- 1.
- 2.

  ABOUT:ME • Kieran Jacobsen •HP Enterprise Services – Engineer/Architect • Microsoft/Automation/Security focus • Twitter: @Kjacobsen • Blog: Aperturescience.su

- 3.

  OUTLINE • PowerShell asan attack platform • PowerShell malware • PowerShell Remoting & WinRM • PowerShell security, and bypassing that security • Defence

- 4.

  CHALLENGE • Move fromsocial engineered workstation to domain controller • Where possible use only PowerShell code • Demo environment will be a “corporate like” environment

- 5.

  ADVANTAGES AS ANATTACK PLATFORM • Code is very easy to develop • Windows integration • Plenty of remote execution options • Designed for automation against 1 – 10000000 devices • Limited security model • Antivirus products are no real concern/limitation • Scripts can be easily hidden from administrators • Installed by DEFAULT

- 6.

  REAL WORLD POWERSHELLMALWARE • Prior to March 2014, only a few minor instances • PowerWorm: • Infect’s Word and Excel documents, initial infection via macro in .doc/.xls • First spotted by TrendMicro, analysis and rewrite by Matt Graeber (@Mattifestation) • PoshKoder/PoshCoder: • PowerWorm crossed with CryptoLocker • Bitcoin ransom

- 7.

  MY POWERSHELL MALWARE •Single Script – SystemInformation.ps1 • Runs as a schedule task, every 5 minutes • Script: • Collects system information and more • Connects to C2 infrastructure, downloads a task list and executes tasks • Executes each task, if successful, task will not be rerun • Tasks can be restricted to individual computers

- 8.
- 9.

  WINDOWS POWERSHELL REMOTINGAND WINRM • PowerShell Remoting is based upon WinRM, Microsoft’s WS-Management implementation • Supports execution in 3 ways: • Remote enabled commands • Remotely executed script blocks • Remote sessions • Security Model = Trusted Devices + User Credentials • WinRM is required for the Windows Server Manager • WinRM is enabled by DEFAULT on Windows 2012(R2) Server • WinRM is allowed through Windows Firewall on all network profiles!

- 10.
- 11.

  POWERSHELL SECURITY FEATURES •Administrative rights • UAC • Code Signing • Local or Remote source using zone.identifier alternate data stream • PowerShell Execution Policy

- 12.

EXECUTION POLICY There are6 states for the execution policy • Unrestricted All scripts can run • Remote Signed No unsigned scripts from the Internet can run • All Signed No unsigned scripts can run • Restricted No scripts are allowed to run • Undefined (Default) If no policy defined, then default to restricted • Bypass Policy processor is bypassed

- 13.

BYPASSING EXECUTION POLICY •Simply ask PowerShell: powershell.exe –executionpolicy unrestricted • Switch the files zone.idenfier back to local: unblock- file yourscript.ps1 • Read the script in and then execute it (may fail depending on script) • Encode the script and use –encodedcommand always works!!!!! • Get/Steal a certificate, sign script, run script

- 14.
- 15.

DEFENCE OF THEDARK ARTS • Restricted/Constrained Endpoints • Change WinRM Listener • Change Windows Firewall settings • Turn it off WinRM • Application whitelisting

- 16.

WINRM, NOT JUSTAN INTERNAL ISSUE By default, Microsoft Azure virtual machines expose HTTPS listener to the Internet.

- 17.

LINKS • Twitter: @kjacobsen •Blog:http://aperturescience.su • Code on GitHub: http://j.mp/1i33Zrk • QuarksPWDump: http://j.mp/1kF30e9 • PowerSploit: http://j.mp/1gJORtF • PowerWorm Analysis: http://j.mp/RzgsHb • PowerBleed: http://j.mp/1jfyILK

- 18.

MORE LINKS • MicrosoftPowerShell/Security Series: • http://j.mp/OOyftt • http://j.mp/1eDYvA4 • http://j.mp/1kF3z7T • http://j.mp/NhSC0X • http://j.mp/NhSEpy • Practical Persistence in PowerShell: http://j.mp/1mU6fQq • Bruteforcing WinRM with PowerShell: http://j.mp/1nBlwX2
