Sign in

redcanaryco / atomic-red-team Public

Notifications Fork 2.8k Star 9.7k

Code Issues 6 Pull requests 4 Actions Wiki Security Insights

atomic-red-team / atomics / T1486 / T1486.md

270 lines (155 loc) · 8.69 KB

T1486 - Data Encrypted for Impact

Description from ATT&CK

> Adversaries may encrypt data on target systems or on large numbers of systems in a network to interrupt availability to system and network resources. They can attempt to render stored data inaccessible by encrypting files or data on local and remote drives and withholding access to a decryption key. This may be done in order to extract monetary compensation from a victim in exchange for decryption or a decryption key (ransomware) or to render data permanently inaccessible in cases where the key is not saved or transmitted.(Citation: US-CERT Ransomware 2016)(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017)(Citation: US-CERT SamSam 2018)
>
> In the case of ransomware, it is typical that common user files like Office documents, PDFs, images, videos, audio, text, and source code files will be encrypted (and often renamed and/or tagged with specific file markers). Adversaries may need to first employ other behaviors, such as File and Directory Permissions Modification or System Shutdown/Reboot, in order to unlock and/or gain access to manipulate these files.(Citation: CarbonBlack Conti July 2020) In some cases, adversaries may encrypt critical system files, disk partitions, and the MBR.(Citation: US-CERT NotPetya 2017)

> To maximize impact on the target organization, malware designed for encrypting data may have worm-like features to propagate across a network by leveraging other attack techniques like Valid Accounts, OS Credential Dumping, and SMB/Windows Admin Shares.(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017) Encryption malware may also leverage Internal Defacement, such as changing victim wallpapers, or otherwise intimidate victims by sending ransom notes or other messages to connected printers (known as "print bombing").(Citation: NHS Digital Egregor Nov 2020)
>
> In cloud environments, storage objects within compromised accounts may also be encrypted. (Citation: Rhino S3 Ransomware Part 1)

Atomic Tests

- Atomic Test #1 - Encrypt files using gpg (Linux)
- Atomic Test #2 - Encrypt files using 7z (Linux)
- Atomic Test #3 - Encrypt files using ccrypt (Linux)
- Atomic Test #4 - Encrypt files using openssl (Linux)
- Atomic Test #5 - PureLocker Ransom Note

Atomic Test #1 - Encrypt files using gpg (Linux)

Uses gpg to encrypt a file

Supported Platforms: Linux

auto_generated_guid: 7b8ce084-3922-4618-8d22-95f996173765

Inputs:

Name	Description	Type	Default Value
pwd_for_encrypted_file	the password that you want for the encrypted file	String	passwd
encrypted_file_path	path to the encrypted file	Path	/tmp/passwd.gpg

input_file_path	path to the file that you want to encrypt	Path	/etc/passwd
encryption_alg	encryption algorithm of the file	String	AES-256

Attack Commands: Run with `bash`!

```
echo "#{pwd_for_encrypted_file}" | $which_gpg --batch --yes --passphrase-fd 0 --ci
```

Cleanup Commands:

```
rm #{encrypted_file_path}
```

Dependencies: Run with `bash`!

Description: Finds where gpg is located

Check Prereq Commands:

```
which_gpg=`which gpg`
```

Get Prereq Commands:

```
(which yum && yum -y install epel-release gpg)||(which apt-get && DEBIAN_FRONTEND=
```

Atomic Test #2 - Encrypt files using 7z (Linux)

Uses 7z to encrypt a file

Supported Platforms: Linux

auto_generated_guid: 53e6735a-4727-44cc-b35b-237682a151ad

Inputs:

Name	Description	Type	Default Value
pwd_for_encrypted_file	the password that you want for the encrypted file	String	passwd
encrypted_file_path	path to the encrypted file	Path	/tmp/passwd.zip
input_file_path	path to the file that you want to encrypt	Path	/etc/passwd

Attack Commands: Run with `bash`!

```
$which_7z a -p#{pwd_for_encrypted_file} #{encrypted_file_path} #{input_file_path}
```

Cleanup Commands:

```
$which_7z e #{encrypted_file_path}
rm #{encrypted_file_path}
```

Dependencies: Run with `bash`!

Description: Finds where 7z is located

Check Prereq Commands:

```
which_7z=`which 7z`
```

Get Prereq Commands:

```

```

Atomic Test #3 - Encrypt files using ccrypt (Linux)

Attempts to encrypt data on target systems as root to simulate an inturruption authentication to target system. If root permissions are not available then attempts to encrypt data within user's home directory.

Supported Platforms: Linux

auto_generated_guid: 08cbf59f-85da-4369-a5f4-049cffd7709f

Inputs:

Name	Description	Type	Default Value
cped_file_path	path where you want your copied file to be	Path	/tmp/passwd
root_input_file_path	path to the file that you want to be encrypted if you are root user	Path	/etc/passwd
user_input_file_path	path to file that you want to be encrypted if you are normal user	Path	~/.bash_history
impact_command	command to show impact of encryption	String	sudo su

Attack Commands: Run with `bash`!

```
if [[ $USER == "root" ]]; then $which_ccencrypt #{root_input_file_path}; file #{ro
```

Cleanup Commands:

```
if [[ $USER == "root" ]]; then mv #{cped_file_path} #{root_input_file_path}; else
```

Dependencies: Run with `bash`!

Description: Finds where ccencrypt and ccdecrypt is located and copies input file

Check Prereq Commands:

```
which_ccencrypt=`which ccencrypt`
which_ccdecrypt=`which ccdecrypt`
if [[ $USER == "root" ]]; then cp #{root_input_file_path} #{cped_file_path}; else
```

Get Prereq Commands:

```
(which yum && yum -y install epel-release ccrypt)||(which apt-get && DEBIAN_FRONTEI
```

Atomic Test #4 - Encrypt files using openssl (Linux)

Uses openssl to encrypt a file

Supported Platforms: Linux

auto_generated_guid: 142752dc-ca71-443b-9359-cf6f497315f1

Inputs:

Name	Description	Type	Default Value
private_key_path	path to the private key	Path	/tmp/key.pem
public_key_path	path to the public key	Path	/tmp/pub.pem
encryption_bit_size	size of the bit of encryption	Integer	2048
encrypted_file_path	path to the encrypted file	Path	/tmp/passwd.zip
input_file_path	path to the file that you want to encrypt	Path	/etc/passwd

Attack Commands: Run with `bash`!

```
$which_openssl genrsa -out #{private_key_path} #{encryption_bit_size}
$which_openssl rsa -in #{private_key_path} -pubout -out #{public_key_path}
$which_openssl rsautl -encrypt -inkey #{public_key_path} -pubin -in #{input_file_p
```

Cleanup Commands:

```
$which_openssl rsautl -decrypt -inkey #{private_key_path} -in #{encrypted_file_pat
rm #{encrypted_file_path}
```

Preview Code Blame Raw

Dependencies: Run with `bash`!

Description: Finds where openssl is located

Check Prereq Commands:

```
which_openssl=`which openssl`
```

Get Prereq Commands:

```

```

Atomic Test #5 - PureLocker Ransom Note

building the IOC (YOUR_FILES.txt) for the PureLocker ransomware https://www.bleepingcomputer.com/news/security/purelocker-ransomware-can-lock-files-on-windows-linux-and-macos/

Supported Platforms: Windows

auto_generated_guid: 649349c7-9abf-493b-a7a2-b1aa4d141528

Attack Commands: Run with `command_prompt`! Elevation Required (e.g. root or admin)

```
echo T1486 - Purelocker Ransom Note > %USERPROFILE%\Desktop\YOUR_FILES.txt
```

Cleanup Commands:

```
del %USERPROFILE%\Desktop\YOUR_FILES.txt >nul 2>&1
```