macOS Security Compliance

macOS 14.0

Security Configuration - MacOSRules

Sonoma Guidance, Revision 2.0 (2024-04-24)

Table of Contents

Chapter 1. Foreword

The macOS Security Compliance Project is an open source effort to provide a programmatic approach to generating security guidance. The configuration settings in this document were derived from National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53, *Security and Privacy Controls for Information Systems and Organizations*, Revision 5.

This project can be used as a resource to easily create customized security baselines of technical security controls by leveraging a library of atomic actions which are mapped to the compliance requirements defined in NIST SP 800-53 (Rev. 5). It can also be used to develop customized guidance to meet the particular cybersecurity needs of any organization.

The objective of this effort was to simplify and radically accelerate the process of producing up-to-date macOS security guidance that is also accessible to any organization and tailorable to meet each organization's specific security needs.

Any and all risk based decisions to tailor the content produced by this project in order to meet the needs of a specific organization shall be approved by the responsible Information System Owner (ISO) and Authorizing Official (AO) and formally documented in their System Security Plan (SSP). While the project attempts to provide settings to meet compliance requirements, it is recommended that each rule be reviewed by your organization's Information System Security Officer (ISSO) prior to implementation.

Chapter 2. Scope

This guide describes the actions to take when securing a macOS 14.0 system against the MacOSRules (Tailored from CIS_LVL1) security baseline.

Chapter 3. Authors

macOS Security Compliance Project

The CIS Benchmarks are referenced with the permission and support of the Center for Internet Security® (CIS®)

Edward Byrd	Center for Internet Security
Ron Colvin	Center for Internet Security
Allen Golbig	Jamf

Chapter 4. Acronyms and Definitions

Table 1. Acronyms and Abbreviations

AES	Advanced Encryption Standard
ABM	Apple Business Manager
AFP	Apple Filing Protocol
ALF	Application Layer Firewall
AO	Authorizing Official
API	Application Programming Interface
ARD	Apple Remote Desktop
CA	Certificate Authority
CIS	Center for Internet Security
CMMC	Cybersecurity Maturity Model Certification
CNSSI	Committee on National Security Systems
CRL	Certificate Revocation List
DISA	Defense Information Systems Agency
DMA	Direct Memory Access
FISMA	Federal Information Security Modernization Act
FPKI	Federal Public Key Infrastructure
IR	Infrared
ISO	Information System Owner
ISSO	Information System Security Officer
MDM	Mobile Device Management
NASA	National Aeronautics and Space Administration
NFS	Network File System
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OCSP	Online Certificate Status Protocol
ODV	Organization Defined Values
OS	Operating System
PF	Packet Filter
PIV	Personal Identity Verification
PIV-M	Personal Identity Verification Mandatory
PKI	Public Key Infrastructure
RBD	Risk Based Decision

SIP	System Integrity Protection
SMB	Server Message Block
SSH	Secure Shell
SSP	System Security Plan
STIG	Security Technical Implementation Guide
UAMDM	User Approved MDM
UUCP	Unix-to-Unix Copy Protocol

Table 2. Definitions

Baseline	A baseline is a predefined set of controls (also referred to as "a catalog" of settings) that address the protection needs of an organization’s information systems. A baseline serves as a starting point for the creation of security benchmarks.
Benchmark	Benchmarks are a defined list of settings with values that an organization has defined.

Chapter 5. Applicable Documents

5.1. Government Documents

Table 3. National Institute of Standards and Technology (NIST)

Document Number or Descriptor	Document Title
NIST Special Publication 800-53 Rev 5	*NIST Special Publication 800-53 Rev 5*
NIST Special Publication 800-63	*NIST Special Publication 800-63*
NIST Special Publication 800-171	*NIST Special Publication 800-171 Rev 2*
NIST Special Publication 800-219	*NIST Special Publication 800-219 Rev 1*

Table 4. Defense Information Systems Agency (DISA)

Document Number or Descriptor	Document Title
STIG Ver 1, Rel 1	*Apple macOS 14 (Sonoma) STIG*

Table 5. Cybersecurity Maturity Model Certification (CMMC)

Document Number or Descriptor	Document Title
CMMC Model Overview v2.0	*Cybersecurity Maturity Model Certification (CMMC) Model Overview v2.0*

Table 6. Committee on National Security Systems (CNSS)

Document Number or Descriptor	Document Title
CNSSI No. 1253	*Security Categorization and Control Selection for National Security Systems*

5.2. Non-Government Documents

Table 7. Apple

Document Number or Descriptor	Document Title
Apple Platform Security Guide	*Apple Platform Security*
Apple Platform Deployment	*Apple Platform Deployment*
Apple Platform Certifications	*Apple Platform Certifications*
Profile-Specific Payload Keys	*Profile-Specific Payload Keys*

Table 8. Center for Internet Security

Document Number or Descriptor	Document Title
Apple macOS 14.0	*CIS Apple macOS 14.0 Benchmark version 1.0.0*

Chapter 6. macOS

This section contains the configuration and enforcement of operating system settings.

> The check/fix commands outlined in this section *MUST* be run by a user with elevated privileges.

6.1. Disable AirDrop

AirDrop *MUST* be disabled to prevent file transfers to or from unauthorized devices. AirDrop allows users to share and receive files from other nearby Apple devices.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.applicationaccess')\
.objectForKey('allowAirDrop').js
EOS
```

If the result is not **false**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.applicationaccess) payload type:

```
<key>allowAirDrop</key>
<false/>
```

ID	os_airdrop_disable	
References	**800-53r5**	• AC-20 • AC-3 • CM-7, CM-7(1)
	CIS Benchmark	• 2.3.1.1 (level 1)
	CIS Controls V8	• 4.1 • 4.8 • 6.7
	CCE	• CCE-92756-6

6.2. Must Use an Approved Antivirus Program

An approved antivirus product *MUST* be installed and configured to run.

Malicious software can establish a base on individual desktops and servers. Employing an automated mechanism to detect this type of software will aid in elimination of the software from the operating system.'

To check the state of the system, run the following command(s):

```
/bin/launchctl list | /usr/bin/grep -cE "(com.apple.XprotectFramework.PluginService
$|com.apple.XProtect.daemon.scan$)"
```

If the result is not **2**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/bin/launchctl load -w
/Library/Apple/System/Library/LaunchDaemons/com.apple.XProtect.daemon.scan.plist
/bin/launchctl load -w
/Library/Apple/System/Library/LaunchDaemons/com.apple.XprotectFramework.PluginServ
ice.plist
```

These services cannot be unloaded or loaded while System Integrity Protection (SIP) is enabled.

ID	os_anti_virus_installed	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.10 (level 1)
	CIS Controls V8	• 10.5 • 10.1 • 10.2
	CCE	• CCE-92758-2

6.3. Enable Authenticated Root

Authenticated Root *MUST* be enabled.

When Authenticated Root is enabled the macOS is booted from a signed volume that is

cryptographically protected to prevent tampering with the system volume.

Authenticated Root is enabled by default on macOS systems.

If more than one partition with macOS is detected, the csrutil command will hang awaiting input.

To check the state of the system, run the following command(s):

```
/usr/bin/csrutil authenticated-root | /usr/bin/grep -c 'enabled'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/csrutil authenticated-root enable
```

To re-enable "Authenticated Root", boot the affected system into "Recovery" mode, launch "Terminal" from the "Utilities" menu, and run the command.

ID	os_authenticated_root_enable	
References	**800-53r5**	• AC-3 • CM-5 • MA-4(1) • SC-34 • SI-7, SI-7(6)
	CIS Benchmark	• 5.1.4 (level 1)
	CIS Controls V8	• 3.6 • 3.11
	CCE	• CCE-92764-0

6.4. Enforce Installation of XProtect Remediator and Gatekeeper Updates Automatically

Software Update *MUST* be configured to update XProtect Remediator and Gatekeeper automatically.

This setting enforces definition updates for XProtect Remediator and Gatekeeper; with this setting in place, new malware and adware that Apple has added to the list of malware or untrusted software will not execute. These updates do not require the computer to be restarted.

https://support.apple.com/en-us/HT207005

Software update will automatically update XProtect Remediator and Gatekeeper by default in the macOS.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.SoftwareUpdate')\
.objectForKey('ConfigDataInstall').js
EOS
```

If the result is not **true**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.SoftwareUpdate) payload type:

```
<key>ConfigDataInstall</key>
<true/>
```

ID	os_config_data_install_enforce	
References	**800-53r5**	• SI-2(5) • SI-3
	CIS Benchmark	• 1.6 (level 1)
	CIS Controls V8	• 7.3 • 7.4 • 7.7
	CCE	• CCE-92776-4

6.5. Enable Firewall Logging

Firewall logging *MUST* be enabled.

Firewall logging ensures that malicious network activity will be logged to the system.

> The firewall data is logged to Apple's Unified Logging with the subsystem `com.apple.alf` and the data is marked as private. In order to enable private data, review the `com.apple.alf.private_data.mobileconfig` file in the project's `includes` folder.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
function run() {
  let pref1 = $.NSUserDefaults.alloc.initWithSuiteName('com.apple.security.firewall')\
  .objectForKey('EnableLogging').js
  let pref2 = $.NSUserDefaults.alloc.initWithSuiteName('com.apple.security.firewall')\
  .objectForKey('LoggingOption').js
  if ( pref1 == true && pref2 == "detail" ){
    return("true")
  } else {
    return("false")
  }
}
EOS
```

If the result is not **true**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.security.firewall) payload type:

```
<key>EnableLogging</key>
<true/>
<key>LoggingOption</key>
<string>detail</string>
```

ID	os_firewall_log_enable

References	**800-53r5**	• AU-12 • SC-7
	CIS Benchmark	• 3.6 (level 1)
	CIS Controls V8	• 4.5 • 8.2 • 8.5
	CCE	• CCE-92793-9

6.6. Enable Gatekeeper

Gatekeeper *MUST* be enabled.

Gatekeeper is a security feature that ensures that applications are digitally signed by an Apple-issued certificate before they are permitted to run. Digital signatures allow the macOS host to verify that the application has not been modified by a malicious third party.

Administrator users will still have the option to override these settings on a case-by-case basis.

To check the state of the system, run the following command(s):

```
/usr/sbin/spctl --status | /usr/bin/grep -c "assessments enabled"
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.systempolicy.control) payload type:

```
<key>EnableAssessment</key>
<true/>
```

ID	os_gatekeeper_enable

References	**800-53r5**	• CM-14 • CM-5 • SI-3 • SI-7(1), SI-7(15)
	CIS Benchmark	• 2.6.5 (level 1)
	CIS Controls V8	• 10.1 • 10.2 • 10.5
	CCE	• CCE-92795-4

6.7. Remove Guest Folder if Present

The guest folder *MUST* be deleted if present.

To check the state of the system, run the following command(s):

```
/bin/ls /Users/ | /usr/bin/grep -c "Guest"
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/bin/rm -Rf /Users/Guest
```

ID	os_guest_folder_removed	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.9 (level 1)
	CIS Controls V8	• 4.1
	CCE	• CCE-92798-8

6.8. Secure User's Home Folders

The system *MUST* be configured to prevent access to other user's home folders.

The default behavior of macOS is to allow all valid users access to the top level of every other user's home folder while restricting access only to the Apple default folders within.

To check the state of the system, run the following command(s):

```
/usr/bin/find /System/Volumes/Data/Users -mindepth 1 -maxdepth 1 -type d ! \( -perm
700 -o -perm 711 \) | /usr/bin/grep -v "Shared" | /usr/bin/grep -v "Guest" |
/usr/bin/wc -l | /usr/bin/xargs
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
IFS=$'\n'
for userDirs in $( /usr/bin/find /System/Volumes/Data/Users -mindepth 1 -maxdepth
1 -type d ! \( -perm 700 -o -perm 711 \) | /usr/bin/grep -v "Shared" |
/usr/bin/grep -v "Guest" ); do
  /bin/chmod og-rwx "$userDirs"
done
unset IFS
```

ID	os_home_folders_secure	
References	**800-53r5**	• AC-6
	CIS Benchmark	• 5.1.1 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-92804-4

6.9. Disable the Built-in Web Server

The built-in web server is a non-essential service built into macOS and *MUST* be disabled.

The built in web server service is disabled at startup by default macOS.

To check the state of the system, run the following command(s):

```
/bin/launchctl print-disabled system | /usr/bin/grep -c '"org.apache.httpd" =>
disabled'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/bin/launchctl disable system/org.apache.httpd
```

ID	os_httpd_disable	
References	**800-53r5**	• AC-17 • AC-3
	CIS Benchmark	• 4.2 (level 1)
	CIS Controls V8	• 4.1 • 4.8
	CCE	• CCE-92805-1

6.10. Configure Install.log Retention to 365

The install.log *MUST* be configured to require records be kept for a organizational defined value before deletion, unless the system uses a central audit record storage facility.

To check the state of the system, run the following command(s):

```
/usr/sbin/aslmanager -dd 2>&1 | /usr/bin/awk '/\/var\/log\/install.log$/ {count++}
/Processing module com.apple.install/,/Finished/ { for (i=1;i<=NR;i++) { if ($i ==
"TTL" && $(i+2) >= 365) { ttl="True" }; if ($i == "MAX") {max="True"}}} END{if (count
> 1) { print "Multiple config files for /var/log/install, manually remove the extra
files"} else if (max == "True") { print "all_max setting is configured, must be
removed" } if (ttl != "True") { print "TTL not configured" } else { print "Yes" }}'
```

If the result is not **Yes**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/sed -i '' "s/\* file \/var\/log\/install.log.*/\* file \/var\/log
\/install.log format='\$\(\(Time\)\(JZ\)\) \$Host \$\(Sender\)\[\$\(PID\\)\]:
\$Message' rotate=utc compress file_max=50M size_only ttl=365/g"
/etc/asl/com.apple.install
```

If there are multiple configuration files in /etc/asl that are set to process the file /var/log/install.log, these files will have to be manually removed.

ID	os_install_log_retention_configure	
References	**800-53r5**	• AU-11 • AU-4
	CIS Benchmark	• 3.3 (level 1)
	CIS Controls V8	• 8.1 • 8.3
	CCE	• CCE-92811-9

6.11. Enforce Enrollment in Mobile Device Management

You *MUST* enroll your Mac in a Mobile Device Management (MDM) software.

User Approved MDM (UAMDM) enrollment or enrollment via Apple Business Manager (ABM)/Apple School Manager (ASM) is required to manage certain security settings. Currently these include:

- Allowed Kernel Extensions
- Allowed Approved System Extensions
- Privacy Preferences Policy Control Payload
- ExtensibleSingleSignOn
- FDEFileVault

In macOS 11, UAMDM grants Supervised status on a Mac, unlocking the following MDM features, which were previously locked behind ABM:

- Activation Lock Bypass
- Access to Bootstrap Tokens
- Scheduling Software Updates
- Query list and delete local users

To check the state of the system, run the following command(s):

```
/usr/bin/profiles status -type enrollment | /usr/bin/awk -F: '/MDM enrollment/ {print
$2}' | /usr/bin/grep -c "Yes (User Approved)"
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Ensure that system is enrolled via UAMDM.

ID	os_mdm_require	
References	**800-53r5**	• CM-2 • CM-6
	CIS Benchmark	• 1.8 (level 1)
	CIS Controls V8	• 4.1 • 5.1
	CCE	• CCE-92824-2

6.12. Enable Apple Mobile File Integrity

Mobile file integrity *MUST* be ebabled.

To check the state of the system, run the following command(s):

```
/usr/sbin/nvram -p | /usr/bin/grep -c "amfi_get_out_of_my_way=1"
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/sbin/nvram boot-args=""
```

ID	os_mobile_file_integrity_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.1.3 (level 1)
	CIS Controls V8	• 2.3 • 2.6
	CCE	• CCE-92828-3

6.13. Disable Network File System Service

Support for Network File Systems (NFS) services is non-essential and, therefore, *MUST* be disabled.

To check the state of the system, run the following command(s):

```
/bin/launchctl print-disabled system | /usr/bin/grep -c '"com.apple.nfsd" => disabled'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/bin/launchctl disable system/com.apple.nfsd
```

The system may need to be restarted for the update to take effect.

ID	os_nfsd_disable	
References	**800-53r5**	• AC-17 • AC-3
	CIS Benchmark	• 4.3 (level 1)
	CIS Controls V8	• 4.1 • 4.8
	CCE	• CCE-92831-7

6.14. Enforce On Device Dictation

Dictation *MUST* be restricted to on device only to prevent potential data exfiltration.

The information system *MUST* be configured to provide only essential capabilities.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.applicationaccess')\
.objectForKey('forceOnDeviceOnlyDictation').js
EOS
```

If the result is not **true**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.applicationaccess) payload type:

```
<key>forceOnDeviceOnlyDictation</key>
<true/>
```

ID	os_on_device_dictation_enforce	
References	**800-53r5**	• AC-20 • CM-7, CM-7(1) • SC-7(10)
	CIS Benchmark	• 2.18.1 (level 1)
	CIS Controls V8	• 4.1 • 4.8
	CCE	• CCE-92841-6

6.15. Remove Password Hint From User Accounts

User accounts *MUST* not contain password hints.

To check the state of the system, run the following command(s):

```
HINT=$(/usr/bin/dscl . -list /Users hint | /usr/bin/awk '{ print $2 }')

if [ -z "$HINT" ]; then
  echo "PASS"
else
  echo "FAIL"
fi
```

If the result is not **PASS**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
for u in $(/usr/bin/dscl . -list /Users UniqueID | /usr/bin/awk '$2 > 500 {print
```

```
$1}'); do
  /usr/bin/dscl . -delete /Users/$u hint
done
```

ID	os_password_hint_remove	
References	**800-53r5**	• IA-6
	CIS Benchmark	• 2.11.1 (level 1)
	CIS Controls V8	• 5.2
	CCE	• CCE-92844-0

6.16. Disable Power Nap

Power Nap *MUST* be disabled.

Power Nap allows your Mac to perform actions while a Mac is asleep. This can interfere with USB power and may cause devices such as smartcards to stop functioning until a reboot and must therefore be disabled on all applicable systems.

The following Macs support Power Nap:

- MacBook (Early 2015 and later)
- MacBook Air (Late 2010 and later)
- MacBook Pro (all models with Retina display)
- Mac mini (Late 2012 and later)
- iMac (Late 2012 and later)
- Mac Pro (Late 2013 and later)

To check the state of the system, run the following command(s):

```
/usr/bin/pmset -g custom | /usr/bin/awk '/powernap/ { sum+=$2 } END {print sum}'
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/pmset -a powernap 0
```

ID	os_power_nap_disable	
References	**800-53r5**	• CM-7, CM-7(1)
	CIS Benchmark	• 2.9.2 (level 1)
	CIS Controls V8	• 4.1 • 4.8
	CCE	• CCE-92853-1

6.17. Disable Root Login

To assure individual accountability and prevent unauthorized access, logging in as root at the login window *MUST* be disabled.

The macOS system *MUST* require individuals to be authenticated with an individual authenticator prior to using a group authenticator, and administrator users *MUST* never log in directly as root.

To check the state of the system, run the following command(s):

```
/usr/bin/dscl . -read /Users/root UserShell 2>&1 | /usr/bin/grep -c "/usr/bin/false"
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/dscl . -create /Users/root UserShell /usr/bin/false
```

ID	os_root_disable	
References	**800-53r5**	• IA-2, IA-2(5)
	CIS Benchmark	• 5.6 (level 1)
	CIS Controls V8	• 5.4
	CCE	• CCE-92875-4

6.18. Ensure Advertising Privacy Protection in Safari Is Enabled

Allow privacy-preserving measurement of ad effectiveness *MUST* be enabled in Safari.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c
'"WebKitPreferences.privateClickMeasurementEnabled" = 1' | /usr/bin/awk '{ if ($1 >=
1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>WebKitPreferences.privateClickMeasurementEnabled</key>
<true/>
```

ID	os_safari_advertising_privacy_protection_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.6 (level 1)
	CIS Controls V8	• 9.1
	CCE	• CCE-92876-2

6.19. Disable Automatic Opening of Safe Files in Safari

Open "safe" files after downloading *MUST* be disabled in Safari.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c 'AutoOpenSafeDownloads = 0' |
/usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>AutoOpenSafeDownloads</key>
<false/>
```

ID	os_safari_open_safe_downloads_disable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.1 (level 1)
	CIS Controls V8	• 9.1 • 9.6
	CCE	• CCE-92877-0

6.20. Ensure Pop-Up Windows are Blocked in Safari

Safari *MUST* be configured to block Pop-Up windows.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c 'safariAllowPopups = 0' |
/usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>safariAllowPopups</key>
<false/>
```

ID	os_safari_popups_disabled

References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.9 (level 1)
	CIS Controls V8	• 9.1
	CCE	• CCE-93014-9

6.21. Ensure Prevent Cross-site Tracking in Safari Is Enabled

Prevent cross-site tracking *MUST* be enabled in Safari.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -cE
'"WebKitPreferences.storageBlockingPolicy" = 1|"WebKitStorageBlockingPolicy" =
1|"BlockStoragePolicy" =2' | /usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print
"0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>WebKitPreferences.storageBlockingPolicy</key>
<integer>1</integer>
<key>WebKitStorageBlockingPolicy</key>
<integer>1</integer>
<key>BlockStoragePolicy</key>
<integer>2</integer>
```

ID	os_safari_prevent_cross-site_tracking_enable

References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.4 (level 1)
	CIS Controls V8	• 9.1 • 9.3
	CCE	• CCE-92878-8

6.22. Ensure Show Full Website Address in Safari Is Enabled

Show full website address *MUST* be enabled in Safari.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c 'ShowFullURLInSmartSearchField = 1'
| /usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>ShowFullURLInSmartSearchField</key>
<true/>
```

ID	os_safari_show_full_website_address_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.7 (level 1)
	CIS Controls V8	• 9.1
	CCE	• CCE-92879-6

6.23. Ensure Show Safari shows the Status Bar is Enabled

Safari *MUST* be configured to show the status bar.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c 'ShowOverlayStatusBar = 1' |
/usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>ShowOverlayStatusBar</key>
<true/>
```

ID	os_safari_show_status_bar_enabled	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.11 (level 1)
	CIS Controls V8	• 9.1
	CCE	• CCE-93015-6

6.24. Ensure Warn When Visiting A Fraudulent Website in Safari Is Enabled

Warn when visiting a fraudulent website *MUST* be enabled in Safari.

To check the state of the system, run the following command(s):

```
/usr/bin/profiles -P -o stdout | /usr/bin/grep -c 'WarnAboutFraudulentWebsites = 1' |
/usr/bin/awk '{ if ($1 >= 1) {print "1"} else {print "0"}}'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Safari) payload type:

```
<key>WarnAboutFraudulentWebsites</key>
<true/>
```

ID	os_safari_warn_fraudulent_website_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.3.3 (level 1)
	CIS Controls V8	• 9.1 • 9.3
	CCE	• CCE-92880-4

6.25. Enable Show All Filename Extensions

Show all filename extensions *MUST* be enabled in the Finder.

The check and fix are for the currently logged in user. To get the currently logged in user, run the following.

```
CURRENT_USER=$( /usr/sbin/scutil <<< "show State:/Users/ConsoleUser" |
/usr/bin/awk '/Name :/ && ! /loginwindow/ { print $3 }' )
```

To check the state of the system, run the following command(s):

```
/usr/bin/sudo -u "$CURRENT_USER" /usr/bin/defaults read .GlobalPreferences
AppleShowAllExtensions 2>/dev/null
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/sudo -u "$CURRENT_USER" /usr/bin/defaults write /Users/"$CURRENT_USER
```

```
"/Library/Preferences/.GlobalPreferences AppleShowAllExtensions -bool true
```

ID	os_show_filename_extensions_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.1.1 (level 1)
	CIS Controls V8	• 2.3
	CCE	• CCE-92888-7

6.26. Ensure System Integrity Protection is Enabled

System Integrity Protection (SIP) *MUST* be enabled.

SIP is vital to protecting the integrity of the system as it prevents malicious users and software from making unauthorized and/or unintended modifications to protected files and folders; ensures the presence of an audit record generation capability for defined auditable events for all operating system components; protects audit tools from unauthorized access, modification, and deletion; restricts the root user account and limits the actions that the root user can perform on protected parts of the macOS; and prevents non-privileged users from granting other users direct access to the contents of their home directories and folders.

SIP is enabled by default in macOS.

To check the state of the system, run the following command(s):

```
/usr/bin/csrutil status | /usr/bin/grep -c 'System Integrity Protection status:
enabled.'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/csrutil enable
```

To reenable "System Integrity Protection", boot the affected system into "Recovery" mode, launch "Terminal" from the "Utilities" menu, and run the command.

ID	os_sip_enable	
References	**800-53r5**	• AC-3 • AU-9, AU-9(3) • CM-5, CM-5(6) • SC-4 • SI-2 • SI-7
	CIS Benchmark	• 5.1.2 (level 1)
	CIS Controls V8	• 2.3 • 2.6 • 10.5
	CCE	• CCE-92889-5

6.27. Ensure Software Update Deferment Is Less Than or Equal to 30 Days

Software updates *MUST* be deferred for 30 days or less.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
function run() {
  let timeout = ObjC.unwrap(
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.applicationaccess')\
.objectForKey('enforcedSoftwareUpdateDelay')) || 0
  if ( timeout <= 30 ) {
    return("true")
  } else {
    return("false")
  }
}
EOS
```

If the result is not **true**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.applicationaccess) payload type:

```
<key>enforcedSoftwareUpdateDelay</key>
<integer>30</integer>
```

ID	os_software_update_deferral	
References	**800-53r5**	• N/A
	CIS Benchmark	• 1.7 (level 1)
	CIS Controls V8	• 7.3 • 7.4
	CCE	• CCE-92893-7

6.28. Configure Sudo Timeout Period to 0

The file /etc/sudoers *MUST* include a timestamp_timeout of 0.

To check the state of the system, run the following command(s):

```
/usr/bin/sudo /usr/bin/sudo -V | /usr/bin/grep -c "Authentication timestamp timeout:
0.0 minutes"
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/find /etc/sudoers* -type f -exec sed -i '' '/timestamp_timeout/d' '{}' \;
/bin/echo "Defaults timestamp_timeout=0" >> /etc/sudoers.d/mscp
```

ID	os_sudo_timeout_configure	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.4 (level 1)
	CIS Controls V8	• 4.3
	CCE	• CCE-92908-3

6.29. Configure Sudoers Timestamp Type

The file /etc/sudoers *MUST* be configured to not include a timestamp_type of global or ppid and be configured for timestamp record types of tty.

This rule ensures that the "sudo" command will prompt for the administrator's password at least once in each newly opened terminal window. This prevents a malicious user from taking advantage of an unlocked computer or an abandoned logon session by bypassing the normal password prompt requirement.

To check the state of the system, run the following command(s):

```
/usr/bin/sudo /usr/bin/sudo -V | /usr/bin/awk -F": " '/Type of authentication
timestamp record/{print $2}'
```

If the result is not **tty**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/find /etc/sudoers* -type f -exec sed -i '' '/timestamp_type/d;
/!tty_tickets/d' '{}' \;
```

ID	os_sudoers_timestamp_type_configure	
References	**800-53r5**	• CM-5(1) • IA-11
	CIS Benchmark	• 5.5 (level 1)
	CIS Controls V8	• 4.3
	CCE	• CCE-92909-1

6.30. Ensure Appropriate Permissions Are Enabled for System Wide Applications

Applications in the System Applications Directory (/Applications) *MUST* not be world-writable.

To check the state of the system, run the following command(s):

```
/usr/bin/find /Applications -iname "*\.app" -type d -perm -2 -ls | /usr/bin/wc -l |
```

```
/usr/bin/xargs
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
IFS=$'\n'
for apps in $( /usr/bin/find /Applications -iname "*\.app" -type d -perm -2 ); do
  /bin/chmod -R o-w "$apps"
done
```

ID	os_system_wide_applications_configure	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.1.5 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-92911-7

6.31. Ensure Secure Keyboard Entry Terminal.app is Enabled

Secure keyboard entry *MUST* be enabled in Terminal.app.

To check the state of the system, run the following command(s):

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.Terminal')\
.objectForKey('SecureKeyboardEntry').js
EOS
```

If the result is not **true**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

Create a configuration profile containing the following keys in the (com.apple.Terminal) payload type:

```
<key>SecureKeyboardEntry</key>
<true/>
```

ID	os_terminal_secure_keyboard_enable	
References	**800-53r5**	• N/A
	CIS Benchmark	• 6.4.1 (level 1)
	CIS Controls V8	• 4.8
	CCE	• CCE-92912-5

6.32. Ensure Time Offset Within Limits

The macOS system time *MUST* be monitored to not drift more than four minutes and thirty seconds.

To check the state of the system, run the following command(s):

```
/usr/bin/sntp $(/usr/sbin/systemsetup -getnetworktimeserver | /usr/bin/awk '{print $4}') | /usr/bin/awk -F'.' '/\+\/\-/{if (substr($1,2) >= 270) {print "No"} else {print "Yes"}}'
```

If the result is not **Yes**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/sntp -Ss $(/usr/sbin/systemsetup -getnetworktimeserver | /usr/bin/awk '{print $4}')
```

ID	os_time_offset_limit_configure	
References	**800-53r5**	• N/A
	CIS Benchmark	• 2.3.2.2 (level 1)
	CIS Controls V8	• 8.4
	CCE	• CCE-92915-8

6.33. Disable Login to Other User's Active and Locked Sessions

The ability to log in to another user's active or locked session *MUST* be disabled.

macOS has a privilege that can be granted to any user that will allow that user to unlock active user's sessions. Disabling the admins and/or user's ability to log into another user's active and locked session prevents unauthorized persons from viewing potentially sensitive and/or personal information.

> Configuring this setting will change the user experience and disable TouchID from unlocking the screensaver. To restore the user experience and allow TouchID to unlock the screensaver, you can run `/usr/bin/sudo /usr/bin/defaults write /Library/Preferences/com.apple.loginwindow screenUnlockMode -int 1`. This setting can also be deployed with a configuration profile.

To check the state of the system, run the following command(s):

```
/usr/bin/security authorizationdb read system.login.screensaver 2>&1 | /usr/bin/grep
-c '<string>use-login-window-ui</string>'
```

If the result is not **1**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
/usr/bin/security authorizationdb write system.login.screensaver "use-login-
window-ui"
```

ID	os_unlock_active_user_session_disable	
References	**800-53r5**	• IA-2, IA-2(5)
	CIS Benchmark	• 5.7 (level 1)
	CIS Controls V8	• 4.3
	CCE	• CCE-92919-0

6.34. Ensure No World Writable Files Exist in the System Folder

Folders in /System/Volumes/Data/System *MUST* not be world-writable.

To check the state of the system, run the following command(s):

```
/usr/bin/find /System/Volumes/Data/System -type d -perm -2 -ls | /usr/bin/grep -vE
"downloadDir|locks" | /usr/bin/wc -l | /usr/bin/xargs
```

If the result is not **0**, this is a finding.

Remediation Description

Perform the following to configure the system to meet the requirements:

```
IFS=$'\n'
for sysPermissions in $( /usr/bin/find /System/Volumes/Data/System -type d -perm
-2 | /usr/bin/grep -vE "downloadDir|locks" ); do
  /bin/chmod -R o-w "$sysPermissions"
done
```

ID	os_world_writable_system_folder_configure	
References	**800-53r5**	• N/A
	CIS Benchmark	• 5.1.6 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-92924-0