



macOS Security Compliance

macOS 12.0

Security Configuration - CIS Benchmarks

Monterey Guidance, Revision 2 (2022-03-16)

Table of Contents

1. Foreword	1
2. Scope	2
3. Authors	3
4. Acronyms and Definitions	4
5. Applicable Documents	6
5.1. Government Documents	6
5.2. Non-Government Documents	6
6. Auditing	8
6.1. Configure Audit Log Files to Not Contain Access Control Lists	8
6.2. Configure Audit Log Folder to Not Contain Access Control Lists	9
6.3. Enable Security Auditing	9
6.4. Configure Audit_Control to Not Contain Access Control Lists	10
6.5. Configure Audit_Control Group to Wheel	11
6.6. Configure Audit_Control Owner to Mode 440 or Less Permissive	12
6.7. Configure Audit_Control Owner to Root	13
6.8. Configure Audit Log Files Group to Wheel	13
6.9. Configure Audit Log Files to Mode 440 or Less Permissive	14
6.10. Configure Audit Log Files to be Owned by Root	15
6.11. Configure Audit Log Folders Group to Wheel	16
6.12. Configure Audit Log Folders to be Owned by Root	17
6.13. Configure Audit Log Folders to Mode 700 or Less Permissive	18
6.14. Configure Audit Retention to a Minimum of Sixty Days or One Gigabyte	19
7. macOS	21
7.1. Disable AirDrop	21
7.2. Enable Authenticated Root	22
7.3. Enforce Installation of XProtect, MRT, and Gatekeeper Updates Automatically	23
7.4. Ensure Extensible Firmware Interface Version is Valid	24
7.5. Enable Firewall Logging	25
7.6. Enable Gatekeeper	26
7.7. Remove Guest Folder if Present	27
7.8. Enable DestroyFVKeyOnStandby on Hibernate	28
7.9. Enable Hibernate Mode	29
7.10. Secure User's Home Folders	30
7.11. Disable the Built-in Web Server	31
7.12. Configure Install.log Retention to 365 Days or More	32
7.13. Enable Library Validation	33
7.14. Enable Apple Mobile File Integrity	34
7.15. Disable Network File System Service	34

7.16. Remove Password Hint From User Accounts	35
7.17. Disable Root Login	36
7.18. Disable Automatic Opening of Safe Files in Safari	37
7.19. Enable Show All Filename Extensions	38
7.20. Ensure System Integrity Protection is Enabled	39
7.21. Configure Sudo Timeout Period to Zero	40
7.22. Configure Sudoers Timestamp Type	41
7.23. Configure Sudoers to Authenticate Users on a Per -tty Basis	42
7.24. Ensure Appropriate Permissions Are Enabled for System Wide Applications	43
7.25. Ensure Secure Keyboard Entry Terminal.app is Enabled	44
7.26. Ensure Time Offset Within Limits	45
7.27. Disable Login to Other User's Active and Locked Sessions	46
7.28. Ensure No World Writable Files Exist in the System Folder	46
8. Password Policy	48
8.1. Limit Consecutive Failed Login Attempts to Five	48
8.2. Prohibit Password Reuse for a Minimum of Fifteen Generations	49
8.3. Require a Minimum Password Length of 15 Characters	50
9. System Preferences	52
9.1. Disable Airplay Receiver	52
9.2. Disable Unattended or Automatic Logon to the System	53
9.3. Enable Bluetooth Menu	54
9.4. Disable Bluetooth Sharing	55
9.5. Disable Bluetooth When No Devices are Paired	56
9.6. Disable CD/DVD Sharing	57
9.7. Enforce Critical Security Updates to be Installed	58
9.8. Enforce FileVault	58
9.9. Enable macOS Application Firewall	59
9.10. Enable Firewall Stealth Mode	60
9.11. Disable Guest Access to Shared SMB Folders	61
9.12. Disable the Guest Account	62
9.13. Enforce macOS Updates are Automatically Installed	63
9.14. Disable Internet Sharing	64
9.15. Configure Login Window to Show A Custom Message	65
9.16. Configure Login Window to Prompt for Username and Password	66
9.17. Disable Password Hints	67
9.18. Disable Personalized Advertising	68
9.19. Disable Power Nap	69
9.20. Disable Printer Sharing	70
9.21. Disable Remote Apple Events	71
9.22. Disable Remote Management	71
9.23. Disable Screen Sharing and Apple Remote Desktop	72

9.24. Enforce Session Lock After Screen Saver is Started	73
9.25. Enforce Screen Saver Timeout	74
9.26. Disable Server Message Block Sharing	75
9.27. Enforce Software Update App Update Updates Automatically	76
9.28. Enforce Software Update Downloads Updates Automatically	77
9.29. Enforce Software Update Automatically	78
9.30. Ensure Software Update is Updated and Current	79
9.31. Disable SSH Server for Remote Access Sessions	80
9.32. Require Administrator Password to Modify System-Wide Preferences	81
9.33. Configure macOS to Use an Authorized Time Server	82
9.34. Enable macOS Time Synchronization Daemon (timed)	83
9.35. Ensure Wake for Network Access Is Disabled	84
9.36. Enable Wifi Menu	85
10. Supplemental	86
10.1. CIS Manual Recommendations	86
10.2. FileVault Supplemental	87
10.3. Packet Filter (pf) Supplemental	88
10.4. Password Policy Supplemental	93
10.5. Smartcard Supplemental	96

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.

2. Scope

This guide describes the actions to take when securing a macOS system against the CIS Apple macOS 12.0 Monterey v1.0.0 Benchmark (Level 1)

3. Authors

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

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
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
OS	Operating System
PF	Packet Filter
PIV	Personal Identity Verification
PIV-M	Personal Identity Verification Mandatory
PKI	Public Key Infrastructure
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

5. Applicable Documents

5.1. Government Documents

Table 2. National Institute of Standards and Technology (NIST)

Document Number or Descriptor	Document Title
NIST Special Publication 800-53 Rev 5	<i>NIST Special Publication 800-53 Rev 5</i>
NIST Special Publication 800-63	<i>NIST Special Publication 800-63</i>
NIST Special Publication 800-171	<i>NIST Special Publication 800-171 Rev 2</i>

Table 3. Defense Information Systems Agency (DISA)

Document Number or Descriptor	Document Title
STIG Ver 1, Rel 1	<i>Apple macOS 12 (Monterey) STIG</i>

Table 4. Committee on National Security Systems (CNSS)

Document Number or Descriptor	Document Title
CNSSI No. 1253	<i>Security Categorization and Control Selection for National Security Systems</i>

5.2. Non-Government Documents

Table 5. Apple

Document Number or Descriptor	Document Title
Apple Platform Security Guide	<i>Apple Platform Security</i>
Deployment Reference for Mac	<i>Deployment Reference</i>
Mobile Device Management Settings	<i>Mobile Device Management Settings</i>
Profile-Specific Payload Keys	<i>Profile-Specific Payload Keys</i>
Security Certifications and Compliance Center	<i>Security Certifications and Compliance Center</i>

Table 6. Center for Internet Security

Document Number or Descriptor	Document Title
Apple macOS 12.0	<i>CIS Apple macOS 12.0 Benchmark version 1.0</i>

6. Auditing

This section contains the configuration and enforcement of the OpenBSM settings.



The BSM Audit subsystem has been marked as deprecated by Apple.



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

6.1. Configure Audit Log Files to Not Contain Access Control Lists

The audit log files *MUST* not contain access control lists (ACLs).

This rule ensures that audit information and audit files are configured to be readable and writable only by system administrators, thereby preventing unauthorized access, modification, and deletion of files.

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

```
/bin/ls -le $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{print $1}' | /usr/bin/grep -c ":"
```

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

Remediation Description

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

```
/bin/chmod -RN $(/usr/bin/awk -F: '/^dir/{print $2}' /etc/security/audit_control)
```

ID	audit_acls_files_configure	
References	800-53r5	• AU-9
	CIS Benchmark	• 3.5 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-90851-7

6.2. Configure Audit Log Folder to Not Contain Access Control Lists

The audit log folder *MUST* not contain access control lists (ACLs).

Audit logs contain sensitive data about the system and users. This rule ensures that the audit service is configured to create log folders that are readable and writable only by system administrators in order to prevent normal users from reading audit logs.

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

```
/bin/ls -lde $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{print $1}' | /usr/bin/grep -c ":"
```

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

Remediation Description

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

```
/bin/chmod -N $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}')
```

ID	audit_acls_folders_configure	
References	800-53r5	• AU-9
	CIS Benchmark	• 3.5 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-90852-5

6.3. Enable Security Auditing

The information system *MUST* be configured to generate audit records.

Audit records establish what types of events have occurred, when they occurred, and which users were involved. These records aid an organization in their efforts to establish, correlate, and investigate the events leading up to an outage or attack.

The content required to be captured in an audit record varies based on the impact level of an

organization’s system. Content that may be necessary to satisfy this requirement includes, for example, time stamps, source addresses, destination addresses, user identifiers, event descriptions, success/fail indications, filenames involved, and access or flow control rules invoked.

The information system initiates session audits at system start-up.



Security auditing is enabled by default on macOS.

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

```
/bin/launchctl list | /usr/bin/grep -c com.apple.auditd
```

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 load -w /System/Library/LaunchDaemons/com.apple.auditd.plist
```

ID	audit_auditd_enabled	
References	800-53r5	<ul style="list-style-type: none">• AU-12, AU-12(1), AU-12(3)• AU-14(1)• AU-3, AU-3(1)• AU-8• CM-5(1)• MA-4(1)
	CIS Benchmark	<ul style="list-style-type: none">• 3.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 8.2, 8.5
	CCE	<ul style="list-style-type: none">• CCE-90854-1

6.4. Configure Audit_Control to Not Contain Access Control Lists

/etc/security/audit_control *MUST* not contain Access Control Lists (ACLs).

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

```
/bin/ls -le /etc/security/audit_control | /usr/bin/awk '{print $1}' | /usr/bin/grep -c ":"
```

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

Remediation Description

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

```
/bin/chmod -N /etc/security/audit_control
```

ID	audit_control_acls_configure	
References	800-53r5	<ul style="list-style-type: none">• AU-9
	CIS Benchmark	<ul style="list-style-type: none">• 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 3.3
	CCE	<ul style="list-style-type: none">• CCE-91088-5

6.5. Configure Audit_Control Group to Wheel

/etc/security/audit_control *MUST* have the group set to wheel.

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

```
/bin/ls -dn /etc/security/audit_control | /usr/bin/awk '{print $4}'
```

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/chgrp wheel /etc/security/audit_control
```

ID	audit_control_group_configure	
References	800-53r5 <ul style="list-style-type: none"> • AU-9 	CIS Benchmark <ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 3.3 	
	CCE <ul style="list-style-type: none"> • CCE-91089-3 	

6.6. Configure Audit_Control Owner to Mode 440 or Less Permissive

/etc/security/audit_control *MUST* be configured so that it is readable only by the root user and group wheel.

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

```
/bin/ls -l /etc/security/audit_control | awk '!/-r--r-----|current|total/{print $1}' | /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:

```
/bin/chmod 440 /etc/security/audit_control
```

ID	audit_control_mode_configure	
References	800-53r5 <ul style="list-style-type: none"> • AU-9 	CIS Benchmark <ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 3.3 	
	CCE <ul style="list-style-type: none"> • CCE-91090-1 	

6.7. Configure Audit_Control Owner to Root

/etc/security/audit_control *MUST* have the owner set to root.

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

```
/bin/ls -dn /etc/security/audit_control | /usr/bin/awk '{print $3}'
```

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/chown root /etc/security/audit_control
```

ID	audit_control_owner_configure	
References	800-53r5	• AU-9
	CIS Benchmark	• 3.5 (level 1)
	CIS Controls V8	• 3.3
	CCE	• CCE-91091-9

6.8. Configure Audit Log Files Group to Wheel

Audit log files *MUST* have the group set to wheel.

The audit service *MUST* be configured to create log files with the correct group ownership to prevent normal users from reading audit logs.

Audit logs contain sensitive data about the system and users. If log files are set to be readable and writable only by system administrators, the risk is mitigated.

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

```
/bin/ls -n $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{s+=$4} END {print s}'
```

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/chgrp -R wheel $(/usr/bin/grep '^dir' /etc/security/audit_control |  
/usr/bin/awk -F: '{print $2}')
```

ID	audit_files_group_configure	
References	800-53r5	<ul style="list-style-type: none">• AU-9
	CIS Benchmark	<ul style="list-style-type: none">• 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 3.3
	CCE	<ul style="list-style-type: none">• CCE-90858-2

6.9. Configure Audit Log Files to Mode 440 or Less Permissive

The audit service *MUST* be configured to create log files that are readable only by the root user and group wheel. To achieve this, audit log files *MUST* be configured to mode 440 or less permissive; thereby preventing normal users from reading, modifying or deleting audit logs.

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

```
/bin/ls -l $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F:  
'{print $2}') | /usr/bin/awk '!/r--r-----|current|total/{print $1}' | /usr/bin/wc -l  
| /usr/bin/tr -d ' '
```

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

Remediation Description

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

```
/bin/chmod 440 $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}')/*
```

ID	audit_files_mode_configure	
References	800-53r5	<ul style="list-style-type: none">• AU-9
	CIS Benchmark	<ul style="list-style-type: none">• 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 3.3
	CCE	<ul style="list-style-type: none">• CCE-90859-0

6.10. Configure Audit Log Files to be Owned by Root

Audit log files *MUST* be owned by root.

The audit service *MUST* be configured to create log files with the correct ownership to prevent normal users from reading audit logs.

Audit logs contain sensitive data about the system and users. If log files are set to only be readable and writable by system administrators, the risk is mitigated.

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

```
/bin/ls -n $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{s+=$3} END {print s}'
```

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/chown -R root $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}')/*
```

ID	audit_files_owner_configure	
References	800-53r5 <ul style="list-style-type: none"> • AU-9 	CIS Benchmark <ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 3.3 	
	CCE <ul style="list-style-type: none"> • CCE-90860-8 	

6.11. Configure Audit Log Folders Group to Wheel

Audit log files *MUST* have the group set to wheel.

The audit service *MUST* be configured to create log files with the correct group ownership to prevent normal users from reading audit logs.

Audit logs contain sensitive data about the system and users. If log files are set to be readable and writable only by system administrators, the risk is mitigated.

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

```
/bin/ls -dn $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{print $4}'
```

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/chgrp wheel $(/usr/bin/awk -F : '/^dir/{print $2}' /etc/security/audit_control)
```

ID	audit_folder_group_configure
-----------	------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AU-9
	CIS Benchmark	<ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-90869-9

6.12. Configure Audit Log Folders to be Owned by Root

Audit log files *MUST* be owned by root.

The audit service *MUST* be configured to create log files with the correct ownership to prevent normal users from reading audit logs.

Audit logs contain sensitive data about the system and users. If log files are set to only be readable and writable by system administrators, the risk is mitigated.

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

```
/bin/ls -dn $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}') | /usr/bin/awk '{print $3}'
```

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/chown root $(/usr/bin/awk -F : '/^dir/{print $2}' /etc/security/audit_control)
```

ID	audit_folder_owner_configure
-----------	------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AU-9
	CIS Benchmark	<ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-90870-7

6.13. Configure Audit Log Folders to Mode 700 or Less Permissive

The audit log folder *MUST* be configured to mode 700 or less permissive so that only the root user is able to read, write, and execute changes to folders.

Because audit logs contain sensitive data about the system and users, the audit service *MUST* be configured to mode 700 or less permissive; thereby preventing normal users from reading, modifying or deleting audit logs.

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

```
/usr/bin/stat -f %A $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}')
```

If the result is not **700**, this is a finding.

Remediation Description

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

```
/bin/chmod 700 $(/usr/bin/grep '^dir' /etc/security/audit_control | /usr/bin/awk -F: '{print $2}')
```

ID	audit_folders_mode_configure
-----------	------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AU-9
	CIS Benchmark	<ul style="list-style-type: none"> • 3.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-90871-5

6.14. Configure Audit Retention to a Minimum of Sixty Days or One Gigabyte

The audit service *MUST* be configured to require records be kept for sixty days or longer before deletion, unless the system uses a central audit record storage facility.

When "expire-after" is set to "60d", the audit service will not delete audit logs until the log data is at least sixty days old.

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

```
/usr/bin/awk -F: '/expire-after/{print $2}' /etc/security/audit_control
```

If the result is not **60d** or **1G**, this is a finding.

Remediation Description

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

```
/usr/bin/sed -i.bak 's/^expire-after.*/expire-after:60d or 1G/'
/etc/security/audit_control; /usr/sbin/audit -s
```

ID	audit_retention_configure_sixty_days
-----------	--------------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AU-11 • AU-4
	CIS Benchmark	<ul style="list-style-type: none"> • 3.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 8.3, 8.1
	CCE	<ul style="list-style-type: none"> • CCE-91093-5

7. 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.

7.1. 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	<ul style="list-style-type: none"> • AC-20 • AC-3 • CM-7, CM-7(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.11 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8, 6.7
	CCE	<ul style="list-style-type: none"> • CCE-90898-8

7.2. 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.

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	<ul style="list-style-type: none"> • AC-3 • CM-5 • MA-4(1) • SC-34 • SI-7, SI-7(6)
	CIS Benchmark	<ul style="list-style-type: none"> • 5.1.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-90907-7

7.3. Enforce Installation of XProtect, MRT, and Gatekeeper Updates Automatically

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

This setting enforces definition updates for XProtect, MRT, 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, MRT, 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	<ul style="list-style-type: none">• SI-2(5)• SI-3
	CIS Benchmark	<ul style="list-style-type: none">• 1.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 10.1, 10.2, 10.4
	CCE	<ul style="list-style-type: none">• CCE-90913-5

7.4. Ensure Extensible Firmware Interface Version is Valid

The macOS Extensible Firmware Interface (EFI) *MUST* be checked to ensure it is a known good version from Apple.

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

```
if /usr/sbin/ioreg -w 0 -c AppleSEPManager | /usr/bin/grep -q AppleSEPManager; then
echo "1"; else /usr/libexec/firmwarecheckers/eficheck/eficheck --integrity-check |
/usr/bin/grep -c "No changes detected"; fi
```

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

Remediation Description

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

Install a known good version of macOS.

ID	os_efi_integrity_validated	
References	800-53r5	<ul style="list-style-type: none">N/A
	CIS Benchmark	<ul style="list-style-type: none">2.11 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">2.2
	CCE	<ul style="list-style-type: none">CCE-91102-4

7.5. 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	<ul style="list-style-type: none">• AU-12• SC-7
	CIS Benchmark	<ul style="list-style-type: none">• 3.6 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.5, 8.2, 8.5
	CCE	<ul style="list-style-type: none">• CCE-90924-2

7.6. 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	<ul style="list-style-type: none">• CM-14• CM-5• SI-3• SI-7(1), SI-7(15)
	CIS Benchmark	<ul style="list-style-type: none">• 2.5.2.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 10.1, 10.2, 10.5
	CCE	<ul style="list-style-type: none">• CCE-90926-7

7.7. 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	<div>800-53r5</div> <div>CIS Benchmark</div> <div>CIS Controls V8</div> <div>CCE</div>	<ul style="list-style-type: none"> • N/A • 6.1.5 (level 1) • N/A • CCE-91104-0

7.8. Enable DestroyFVKeyOnStandby on Hibernate

DestroyFVKeyOnStandby on hibernate *MUST* be enabled.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.MCX')\
.objectForKey('DestroyFVKeyOnStandby').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.MCX) payload type:

```
<key>DestroyFVKeyOnStandby</key>
<true/>
```

ID	os_hibernate_mode_destroyfvkeyonstandby_enable
-----------	--

References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.9 (level 2)
	CIS Controls V8	<ul style="list-style-type: none"> • N/A
	CCE	<ul style="list-style-type: none"> • CCE-91105-7

7.9. Enable Hibernate Mode

Hibernate mode *MUST* be enabled.



Hibernate mode is not fully supported on Apple Silicon devices. This rule is only applicable to Intel devices.

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

```
error_count=0
hibernateStandbyLowValue=$(/usr/bin/pmset -g | /usr/bin/grep standbydelaylow 2>&1 | /usr/bin/awk '{print $2}')
hibernateStandbyHighValue=$(/usr/bin/pmset -g | /usr/bin/grep standbydelayhigh 2>&1 | /usr/bin/awk '{print $2}')
hibernateStandbyThreshValue=$(/usr/bin/pmset -g | /usr/bin/grep highstandbythreshold 2>&1 | /usr/bin/awk '{print $2}')
hibernateMode=$(/usr/bin/pmset -b -g | /usr/bin/grep hibernatemode 2>&1 | /usr/bin/awk '{print $2}')
macType=$(/usr/sbin/system_profiler SPHardwareDataType 2>&1 | /usr/bin/grep -c MacBook)
if [[ "$macType" -ge 0 ]]; then
    if [[ "$hibernateStandbyLowValue" == "" ]] || [[ "$hibernateStandbyLowValue" -gt 600 ]]; then
        ((error_count++))
    fi
    if [[ "$hibernateStandbyHighValue" == "" ]] || [[ "$hibernateStandbyHighValue" -gt 600 ]]; then
        ((error_count++))
    fi
    if [[ "$hibernateStandbyThreshValue" == "" ]] || [[ "$hibernateStandbyThreshValue" -lt 90 ]]; then
        ((error_count++))
    fi
fi
echo "$error_count"
```

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 standbydelayhigh 600  
/usr/bin/pmset -a standbydelaylow 600  
/usr/bin/pmset -a highstandbythreshold 90
```

ID	os_hibernate_mode_enable	
References	800-53r5	<ul style="list-style-type: none">• N/A
	CIS Benchmark	<ul style="list-style-type: none">• 5.9 (level 2)
	CIS Controls V8	<ul style="list-style-type: none">• N/A
	CCE	<ul style="list-style-type: none">• CCE-91106-5

7.10. 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 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 -1 |  
/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 $( /usr/bin/find /System/Volumes/Data/Users -mindepth 1 -maxdepth
1 -type d -perm -1 | /usr/bin/grep -v "Shared" | /usr/bin/grep -v "Guest" ); do
  /bin/chmod og-rwx "$userDirs"
done
unset IFS
```

ID	os_home_folders_secure	
References	800-53r5	• AC-6
	CIS Benchmark	• 5.1.1 (level 1)
	CIS Controls V8	• N/A
	CCE	• CCE-90931-7

7.11. 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" => true'
```

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	<div>800-53r5</div> <ul style="list-style-type: none"> • AC-17 • AC-3 	<div>CIS Benchmark</div> <ul style="list-style-type: none"> • 4.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3, 6.7
	CCE	<ul style="list-style-type: none"> • CCE-90932-5

7.12. Configure Install.log Retention to 365 Days or More

The install.log *MUST* be configured to require records be kept for 365 days or longer 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"} else if
(ttl != "True") { print "TTL not configured" } else if (max == "True") { print "Max
Size is configured, must be removed" } 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	<ul style="list-style-type: none">• AU-11• AU-4	
	CIS Benchmark	<ul style="list-style-type: none">• 3.3 (level 1)	
	CIS Controls V8	<ul style="list-style-type: none">• 8.1, 8.3	
	CCE	<ul style="list-style-type: none">• CCE-91107-3	

7.13. Enable Library Validation

Library validation *MUST* be enabled.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.security.libraryvalidation')\
.objectForKey('DisableLibraryValidation').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.security.libraryvalidation) payload type:

```
<key>DisableLibraryValidation</key>
<false/>
```

ID	os_library_validation_enabled
-----------	-------------------------------

References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.1.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 2.3, 2.6
	CCE	<ul style="list-style-type: none"> • CCE-91108-1

7.14. Enable Apple Mobile File Integrity

Mobile file integrity *MUST* be enabled.

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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.1.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 2.3, 2.6
	CCE	<ul style="list-style-type: none"> • CCE-91109-9

7.15. 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" => true'
```

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	<ul style="list-style-type: none">• AC-17• AC-3
	CIS Benchmark	<ul style="list-style-type: none">• 4.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 3.3, 6.7
	CCE	<ul style="list-style-type: none">• CCE-90956-4

7.16. 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):

```
/usr/bin/dscl . -list /Users hint | /usr/bin/awk '{print $2}' | /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:

```
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	• 5.14 (level 1)
	CIS Controls V8	• 5.2
	CCE	• CCE-91110-7

7.17. 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 <ul style="list-style-type: none"> • IA-2, IA-2(5) 	CIS Benchmark <ul style="list-style-type: none"> • 5.6 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 4.7 	
	CCE <ul style="list-style-type: none"> • CCE-90994-5 	

7.18. 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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 6.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 9
	CCE	<ul style="list-style-type: none"> • CCE-91111-5

7.19. 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=$( scutil <<< "show State:/Users/ConsoleUser" \> | 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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 6.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 2.3
	CCE	<ul style="list-style-type: none"> • CCE-91112-3

7.20. 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 reenble "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	<ul style="list-style-type: none"> • AC-3 • AU-9, AU-9(3) • CM-5, CM-5(6) • SC-4 • SI-2 • SI-7
	CIS Benchmark	<ul style="list-style-type: none"> • 5.18 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 2.6, 3.3, 10.5
	CCE	<ul style="list-style-type: none"> • CCE-91000-0

7.21. Configure Sudo Timeout Period to Zero

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

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

```
/usr/bin/find /etc/sudoers* -type f -exec /usr/bin/grep -E "^Defaults
\s+timestamp_timeout=0" '{}' \; | /usr/bin/wc -l | /usr/bin/xargs
```

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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.3
	CCE	<ul style="list-style-type: none"> • CCE-91116-4

7.22. Configure Sudoers Timestamp Type

The file `/etc/sudoers` *MUST* be configured to not include a timestamp_type of global or ppid.

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/find /etc/sudoers* -type f -exec /usr/bin/grep -E
'^Defaults\s+timestamp_type=global|^Defaults\s+timestamp_type=ppid)' '{}' \; |
/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:

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

ID	os_sudoers_timestamp_type_configure
-----------	-------------------------------------

References	800-53r5	<ul style="list-style-type: none"> • CM-5(1) • IA-11
	CIS Benchmark	<ul style="list-style-type: none"> • 5.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.3
	CCE	<ul style="list-style-type: none"> • CCE-91015-8

7.23. Configure Sudoers to Authenticate Users on a Per-tty Basis

The file `/etc/sudoers` *MUST* be configured to include `tty_tickets`.

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. Without the "tty_tickets" option, all open local and remote logon sessions would be authenticated to use sudo without a password for the duration of the configured password timeout window.

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

```
/usr/bin/find /etc/sudoers* -type f -exec /usr/bin/grep -E "^Defaults\s+\\!tty_tickets"
'{}' \; | /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:

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

ID	os_sudoers_tty_configure
-----------	--------------------------

References	800-53r5	<ul style="list-style-type: none"> • CM-5(1) • IA-11
	CIS Benchmark	<ul style="list-style-type: none"> • 5.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.3
	CCE	<ul style="list-style-type: none"> • CCE-91015-8

7.24. 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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.1.6 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-91117-2

7.25. 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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 2.10 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91118-0

7.26. 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	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 2.2.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 8.4
	CCE	<ul style="list-style-type: none"> • CCE-91119-8

7.27. 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 andlocked session prevents unauthorized persons from viewing potentially sensitive and/or personal information.

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 'use-login-window-ui'
```

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	• 5.11 (level 1)
	CIS Controls V8	• 4.3
	CCE	• CCE-91022-4

7.28. 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 -v "Drop Box" | /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 $( /usr/bin/find /System/Volumes/Data/System -type d -perm -2 | /usr/bin/grep -v "Drop Box" ); do
    /bin/chmod -R o-w "$sysPermissions"
done
```

ID	os_world_writable_system_folder_configure	
References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 5.1.7 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3
	CCE	<ul style="list-style-type: none"> • CCE-91121-4

8. Password Policy

This section contains the configuration and enforcement of settings pertaining to password policies in macOS.



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



The password policy recommendations in the NIST 800-53 (Rev 5) and NIST 800-63B state that complexity rules should be organizationally defined. The values defined are based off of common complexity values. But your organization may define its own password complexity rules.



The settings outlined in this section adhere to the recommendations provided in this document for systems that utilize passwords for local accounts. If systems are integrated with a directory service, local password policies should align with domain password policies to the fullest extent feasible.

8.1. Limit Consecutive Failed Login Attempts to Five

The macOS *MUST* be configured to limit the number of failed login attempts to a maximum of five. When the maximum number of failed attempts is reached, the account *MUST* be locked for a period of time after.

This rule protects against malicious users attempting to gain access to the system via brute-force hacking methods.

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

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

If the result is not 5, 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.mobiledevice.passwordpolicy) payload type:

```
<key>maxFailedAttempts</key>
<integer>5</integer>
```

ID	pwpolicy_account_lockout_enforce_five	
References	800-53r5	• AC-7
	CIS Benchmark	• 5.2.1 (level 1)
	CIS Controls V8	• 6.2
	CCE	• CCE-91122-2

8.2. Prohibit Password Reuse for a Minimum of Fifteen Generations

The macOS *MUST* be configured to enforce a password history of at least fifteen previous passwords when a password is created.

This rule ensures that users are not allowed to re-use a password that was used in any of the fifteen previous password generations.

Limiting password reuse protects against malicious users attempting to gain access to the system via brute-force hacking methods.

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

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

If the result is not 15, 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.mobiledevice.passwordpolicy) payload type:

```
<key>pinHistory</key>
<integer>15</integer>
```

ID	pwpolicy_history_enforce_fifteen	
References	800-53r5	• IA-5(1)
	CIS Benchmark	• 5.2.8 (level 1)
	CIS Controls V8	• 5.2
	CCE	• CCE-91123-0

8.3. Require a Minimum Password Length of 15 Characters

The macOS *MUST* be configured to require a minimum of 15 characters be used when a password is created.

This rule enforces password complexity by requiring users to set passwords that are less vulnerable to malicious users.



The guidance for password based authentication in NIST 800-53 (Rev 5) and NIST 800-63B state that complexity rules should be organizationally defined. The values defined are based off of common complexity values. But your organization may define its own password complexity rules.

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

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

If the result is not 15, 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.mobiledevice.passwordpolicy) payload type:

```
<key>minLength</key>
<integer>15</integer>
```

ID	pwpolicy_minimum_length_enforce	
References	800-53r5	• IA-5(1)
	CIS Benchmark	• 5.2.2 (level 1)
	CIS Controls V8	• 5.2
	CCE	• CCE-91036-4

9. System Preferences

This section contains the configuration and enforcement of the settings within the macOS System Preferences application.



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

9.1. Disable Airplay Receiver

Airplay Receiver allows you to send content from another Apple device to be displayed on the screen as it's being played from your other device.

Support for Airplay Receiver is non-essential and *MUST* be disabled.

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('allowAirPlayIncomingRequests').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>allowAirPlayIncomingRequests</key>
<false/>
```

ID	sysprefs_airplay_receiver_disable
----	-----------------------------------

References	800-53r5	<ul style="list-style-type: none"> • CM-7, CM-7(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.13 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91044-8

9.2. Disable Unattended or Automatic Logon to the System

Automatic logon *MUST* be disabled.

When automatic logons are enabled, the default user account is automatically logged on at boot time without prompting the user for a password. Even if the screen is later locked, a malicious user would be able to reboot the computer and find it already logged in. Disabling automatic logons mitigates this risk.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.loginwindow')\
.objectForKey('com.apple.login.mcx.DisableAutoLoginClient').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.loginwindow) payload type:

```
<key>com.apple.login.mcx.DisableAutoLoginClient</key>
<true/>
```

ID	sysprefs_automatic_login_disable
-----------	----------------------------------

References	800-53r5	<ul style="list-style-type: none"> • IA-2 • IA-5(13)
	CIS Benchmark	<ul style="list-style-type: none"> • 5.7 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.7
	CCE	<ul style="list-style-type: none"> • CCE-91046-3

9.3. Enable Bluetooth Menu

The bluetooth menu *MUST* be enabled.

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

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

If the result is not **18**, 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.controlcenter) payload type:

```
<key>Bluetooth</key>
<integer>18</integer>
```

ID	sysprefs_bluetooth_menu_enable
-----------	--------------------------------

References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 2.1.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.8, 13.9
	CCE	<ul style="list-style-type: none"> • CCE-91124-8

9.4. Disable Bluetooth Sharing

Bluetooth Sharing *MUST* be disabled.

Bluetooth Sharing allows users to wirelessly transmit files between the macOS and Bluetooth-enabled devices, including personally owned cellphones and tablets. A malicious user might introduce viruses or malware onto the system or extract sensitive files via Bluetooth Sharing. When Bluetooth Sharing is disabled, this risk is mitigated.



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

```
CURRENT_USER=$( scutil <<< "show State:/Users/ConsoleUser" \> | 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 -currentHost read
com.apple.Bluetooth PrefKeyServicesEnabled
```

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/sudo -u "$CURRENT_USER" /usr/bin/defaults -currentHost write
com.apple.Bluetooth PrefKeyServicesEnabled -bool false
```

ID	sysprefs_bluetooth_sharing_disable
----	------------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AC-18(4) • AC-3 • CM-7, CM-7(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.7 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 3.3, 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91049-7

9.5. Disable Bluetooth When No Devices are Paired

Bluetooth *MUST* be disabled when no devices are paired.

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

```
isPaired=$(/usr/sbin/system_profiler SPBluetoothDataType 2>/dev/null | /usr/bin/grep
-c 'Connected: Yes')
if [[ "$isPaired" = "0" ]]; then
    powerState=$(/usr/sbin/system_profiler SPBluetoothDataType 2>/dev/null |
/usr/bin/grep -c 'State: On')
    /bin/echo "$powerState"
else
    /bin/echo "0"
fi
```

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/defaults write
/private/var/root/Library/Preferences/com.apple.BTServer.plist defaultPoweredState
off
/usr/bin/killall -HUP bluetoothd
```

ID	sysprefs_bluetooth_unpaired_disable
----	-------------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AC-18, AC-18(3) • SC-8
	CIS Benchmark	<ul style="list-style-type: none"> • 2.1.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.8, 12.6, 13.9
	CCE	<ul style="list-style-type: none"> • CCE-91126-3

9.6. Disable CD/DVD Sharing

CD/DVD Sharing *MUST* be disabled.

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

```
/usr/bin/pgrep -q ODSEAgent; /bin/echo $?
```

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 unload /System/Library/LaunchDaemons/com.apple.ODSEAgent.plist
```

ID	sysprefs_cd_dvd_sharing_disable	
References	800-53r5	<ul style="list-style-type: none"> • CM-7, CM-7(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.6 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91127-1

9.7. Enforce Critical Security Updates to be Installed

Ensure that security updates are installed as soon as they are available from Apple.

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('CriticalUpdateInstall').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>CriticalUpdateInstall</key>
<true/>
```

ID	sysprefs_critical_update_install_enforce	
References	800-53r5	• SI-2
	CIS Benchmark	• 1.5 (level 1)
	CIS Controls V8	• 7.3, 7.4, 7.7
	CCE	• CCE-91051-3

9.8. Enforce FileVault

FileVault *MUST* be enforced.

The information system implements cryptographic mechanisms to protect the confidentiality and integrity of information stored on digital media during transport outside of controlled areas.

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

```
/usr/bin/fdesetup status | /usr/bin/grep -c "FileVault is On."
```

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

Remediation Description

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



See the FileVault supplemental to implement this rule.

ID	sysprefs_filevault_enforce	
References	800-53r5	• SC-28, SC-28(1)
	CIS Benchmark	• 2.5.5.1 (level 1)
	CIS Controls V8	• 3.6, 3.11
	CCE	• CCE-91053-9

9.9. Enable macOS Application Firewall

The macOS Application Firewall is the built-in firewall that comes with macOS, and it *MUST* be enabled.

When the macOS Application Firewall is enabled, the flow of information within the information system and between interconnected systems will be controlled by approved authorizations.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.security.firewall')\
.objectForKey('EnableFirewall').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.security.firewall) payload type:

```
<key>EnableFirewall</key>
<true/>
```

ID	sysprefs_firewall_enable	
References	800-53r5	<ul style="list-style-type: none">• AC-4• CM-7, CM-7(1)• SC-7, SC-7(12)
	CIS Benchmark	<ul style="list-style-type: none">• 2.5.2.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.1, 4.5, 13.1
	CCE	<ul style="list-style-type: none">• CCE-91055-4

9.10. Enable Firewall Stealth Mode

Firewall Stealth Mode *MUST* be enabled.

When stealth mode is enabled, the Mac will not respond to any probing requests, and only requests from authorized applications will still be authorized.



Enabling firewall stealth mode may prevent certain remote mechanisms used for maintenance and compliance scanning from properly functioning. Information System Security Officers (ISSOs) are advised to first fully weigh the potential risks posed to their organization before opting not to enable stealth mode.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.security.firewall')\
.objectForKey('EnableStealthMode').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.security.firewall) payload type:

```
<key>EnableStealthMode</key>
<true/>
```

ID	sysprefs_firewall_stealth_mode_enable	
References	800-53r5	<ul style="list-style-type: none">• CM-7, CM-7(1)• SC-7, SC-7(16)
	CIS Benchmark	<ul style="list-style-type: none">• 2.5.2.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.1, 4.5, 4.8
	CCE	<ul style="list-style-type: none">• CCE-91056-2

9.11. Disable Guest Access to Shared SMB Folders

Guest access to shared Server Message Block (SMB) folders *MUST* be disabled.

Turning off guest access prevents anonymous users from accessing files shared via SMB.

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

```
/usr/bin/defaults read /Library/Preferences/SystemConfiguration/com.apple.smb.server AllowGuestAccess
```

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/sysadminctl -smbGuestAccess off
```

ID	sysprefs_guest_access_smb_disable	
References	800-53r5	• AC-2, AC-2(9)
	CIS Benchmark	• 6.1.4 (level 1)
	CIS Controls V8	• 5.2, 6.2, 6.8
	CCE	• CCE-91059-6

9.12. Disable the Guest Account

Guest access *MUST* be disabled.

Turning off guest access prevents anonymous users from accessing files.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.MCX')\
.objectForKey('DisableGuestAccount').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.MCX) payload type:

```
<key>DisableGuestAccount</key>
<true/>
```

ID	sysprefs_guest_account_disable	
References	<div>800-53r5</div> <div>CIS Benchmark</div> <div>CIS Controls V8</div> <div>CCE</div>	<ul style="list-style-type: none"> • AC-2, AC-2(9) • 6.1.3 (level 1) • 5.2, 5.3, 6.8 • CCE-91060-4

9.13. Enforce macOS Updates are Automatically Installed

Software Update *MUST* be configured to enforce automatic installation of macOS updates is enabled.

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('AutomaticallyInstallMacOSUpdates').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>AutomaticallyInstallMacOSUpdates</key>
<true/>
```

ID	sysprefs_install_macos_updates_enforce
-----------	--

References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 1.6 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 7.3, 7.4
	CCE	<ul style="list-style-type: none"> • CCE-91129-7

9.14. Disable Internet Sharing

If the system does not require Internet sharing, support for it is non-essential and *MUST* be disabled.

The information system *MUST* be configured to provide only essential capabilities. Disabling Internet sharing helps prevent the unauthorized connection of devices, unauthorized transfer of information, and unauthorized tunneling.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.MCX')\
.objectForKey('forceInternetSharingOff').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.MCX) payload type:

```
<key>forceInternetSharingOff</key>
<true/>
```

ID	sysprefs_internet_sharing_disable
-----------	-----------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AC-20 • AC-4
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91063-8

9.15. Configure Login Window to Show A Custom Message

The login window *MUST* be configured to show a custom access warning message.

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

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

If the result is not **Approved message goes here**, 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.loginwindow) payload type:

```
<key>LoginwindowText</key>
<string>Approved message goes here</string>
```

ID	sysprefs_loginwindow_loginwindowtext_enable
----	---

References	800-53r5	<ul style="list-style-type: none"> • N/A
	CIS Benchmark	<ul style="list-style-type: none"> • 6.1.1 (level 1)
	CCE	<ul style="list-style-type: none"> • CCE-91133-9

9.16. Configure Login Window to Prompt for Username and Password

The login window *MUST* be configured to prompt all users for both a username and a password.

By default, the system displays a list of known users on the login window, which can make it easier for a malicious user to gain access to someone else's account. Requiring users to type in both their username and password mitigates the risk of unauthorized users gaining access to the information system.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.loginwindow')\
.objectForKey('SHOWFULLNAME').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.loginwindow) payload type:

```
<key>SHOWFULLNAME</key>
<true/>
```

ID	sysprefs_loginwindow_prompt_username_password_enforce
----	---

References	800-53r5	<ul style="list-style-type: none"> • IA-2
	CIS Benchmark	<ul style="list-style-type: none"> • 6.1.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1
	CCE	<ul style="list-style-type: none"> • CCE-91065-3

9.17. Disable Password Hints

Password hints *MUST* be disabled.

Password hints leak information about passwords that are currently in use and can lead to loss of confidentiality.

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

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

If the result is not **0**, 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.loginwindow) payload type:

```
<key>RetriesUntilHint</key>
<integer>0</integer>
```

ID	sysprefs_password_hints_disable
-----------	---------------------------------

References	800-53r5	<ul style="list-style-type: none"> • IA-6
	CIS Benchmark	<ul style="list-style-type: none"> • 6.1.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1
	CCE	<ul style="list-style-type: none"> • CCE-91067-9

9.18. Disable Personalized Advertising

Ad tracking and targeted ads *MUST* be disabled.

The information system *MUST* be configured to provide only essential capabilities. Disabling ad tracking ensures that applications and advertisers are unable to track users' interests and deliver targeted advertisements.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.AdLib')\
.objectForKey('allowApplePersonalizedAdvertising').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.AdLib) payload type:

```
<key>allowApplePersonalizedAdvertising</key>
<false/>
```

ID	sysprefs_personalized_advertising_disable
-----------	---

References	800-53r5	<ul style="list-style-type: none"> • AC-20 • CM-7, CM-7(1) • SC-7(10)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.5.6 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91068-7

9.19. 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 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	sysprefs_power_nap_disable	
References	800-53r5 <ul style="list-style-type: none"> • CM-7, CM-7(1) 	CIS Benchmark <ul style="list-style-type: none"> • 2.9 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 4.1, 4.8 	
	CCE <ul style="list-style-type: none"> • CCE-91069-5 	

9.20. Disable Printer Sharing

Printer Sharing *MUST* be disabled.

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

```
/usr/sbin/cupctl | /usr/bin/grep -c "_share_printers=0"
```

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

Remediation Description

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

```
/usr/sbin/cupctl --no-share-printers
/usr/bin/lpstat -p | awk '{print $2}' | /usr/bin/xargs -I{} lpadmin -p {} -o
printer-is-shared=false
```

ID	sysprefs_printer_sharing_disable	
References	800-53r5 <ul style="list-style-type: none"> • CM-7, CM-7(1) 	CIS Benchmark <ul style="list-style-type: none"> • 2.4.4 (level 1)
	CIS Controls V8 <ul style="list-style-type: none"> • 4.1, 4.8 	
	CCE <ul style="list-style-type: none"> • CCE-91134-7 	

9.21. Disable Remote Apple Events

If the system does not require Remote Apple Events, support for Apple Remote Events is non-essential and *MUST* be disabled.

The information system *MUST* be configured to provide only essential capabilities. Disabling Remote Apple Events helps prevent the unauthorized connection of devices, the unauthorized transfer of information, and unauthorized tunneling.

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

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

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

Remediation Description

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

```
/usr/sbin/systemsetup -setremoteappleevents off  
/bin/launchctl disable system/com.apple.AEServer
```



Systemsetup with -setremoteappleevents flag will fail unless you grant Full Disk Access to systemsetup or it's parent process. Requires UAMDM.

ID	sysprefs_rae_disable	
References	800-53r5	<ul style="list-style-type: none">• AC-17• AC-3
	CIS Benchmark	<ul style="list-style-type: none">• 2.4.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.1, 4.8
	CCE	<ul style="list-style-type: none">• CCE-91070-3

9.22. Disable Remote Management

Remote Management *MUST* be disabled.

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

```
/usr/libexec/mdmclient QuerySecurityInfo | /usr/bin/grep -c "RemoteDesktopEnabled = 0"
```

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

Remediation Description

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

```
/System/Library/CoreServices/RemoteManagement/ARDAgent.app/Contents/Resources/kick  
start -deactivate -stop
```

ID	sysprefs_remote_management_disable	
References	800-53r5	• CM-7, CM-7(1)
	CIS Benchmark	• 2.4.3 (level 1)
	CIS Controls V8	• 4.1, 4.8
	CCE	• CCE-91135-4

9.23. Disable Screen Sharing and Apple Remote Desktop

Support for both Screen Sharing and Apple Remote Desktop (ARD) is non-essential and *MUST* be disabled.

The information system *MUST* be configured to provide only essential capabilities. Disabling screen sharing and ARD helps prevent the unauthorized connection of devices, the unauthorized transfer of information, and unauthorized tunneling.

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

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

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.screensharing
```

NOTE - This will apply to the whole system

ID	sysprefs_screen_sharing_disable	
References	800-53r5	<ul style="list-style-type: none">• AC-17• AC-3
	CIS Benchmark	<ul style="list-style-type: none">• 2.4.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.1, 4.8
	CCE	<ul style="list-style-type: none">• CCE-91071-1

9.24. Enforce Session Lock After Screen Saver is Started

A screen saver *MUST* be enabled and the system *MUST* be configured to require a password to unlock once the screensaver has been on for a maximum of five seconds.

An unattended system with an excessive grace period is vulnerable to a malicious user.

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

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

If the result is not 5, 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.screensaver) payload type:

```
<key>askForPasswordDelay</key>
<integer>5</integer>
```

ID	sysprefs_screensaver_ask_for_password_delay_enforce	
References	800-53r5	<ul style="list-style-type: none">• AC-11
	CIS Benchmark	<ul style="list-style-type: none">• 5.8 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.7
	CCE	<ul style="list-style-type: none">• CCE-91072-9

9.25. Enforce Screen Saver Timeout

The screen saver timeout *MUST* be set to 20 minutes or a shorter length of time.

This rule ensures that a full session lock is triggered within no more than 20 minutes of inactivity.

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

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

If the result is not **1200**, 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.screensaver) payload type:

```
<key>idleTime</key>
<integer>1200</integer>
```

ID	sysprefs_screensaver_timeout_enforce	
References	800-53r5	<ul style="list-style-type: none">• AC-11• IA-11
	CIS Benchmark	<ul style="list-style-type: none">• 2.3.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.3
	CCE	<ul style="list-style-type: none">• CCE-91074-5

9.26. Disable Server Message Block Sharing

Support for Server Message Block (SMB) file sharing is non-essential and *MUST* be disabled.

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

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

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

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.smbd
```

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

ID	sysprefs_smbd_disable	
References	800-53r5	<ul style="list-style-type: none">• AC-17• AC-3
	CIS Benchmark	<ul style="list-style-type: none">• 2.4.8 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.1, 4.8
	CCE	<ul style="list-style-type: none">• CCE-91076-0

9.27. Enforce Software Update App Update Updates Automatically

Software Update *MUST* be configured to enforce automatic updates of App Updates is enabled.

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('AutomaticallyInstallAppUpdates').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>AutomaticallyInstallAppUpdates</key>
<true/>
```

ID	sysprefs_software_update_app_update_enforce	
References	800-53r5	<ul style="list-style-type: none">• N/A
	CIS Benchmark	<ul style="list-style-type: none">• 1.4 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 7.3, 7.4
	CCE	<ul style="list-style-type: none">• CCE-91138-8

9.28. Enforce Software Update Downloads Updates Automatically

Software Update *MUST* be configured to enforce automatic downloads of updates is enabled.

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('AutomaticDownload').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>AutomaticDownload</key>
<true/>
```

ID	sysprefs_software_update_download_enforce	
References	800-53r5	<ul style="list-style-type: none">• N/A
	CIS Benchmark	<ul style="list-style-type: none">• 1.3 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 7.3, 7.4
	CCE	<ul style="list-style-type: none">• CCE-91139-6

9.29. Enforce Software Update Automatically

Software Update *MUST* be configured to enforce automatic update is enabled.

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('AutomaticCheckEnabled').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>AutomaticCheckEnabled</key>
<true/>
```

ID	sysprefs_software_update_enforce	
References	800-53r5	• SI-2(5)
	CIS Benchmark	• 1.2 (level 1)
	CIS Controls V8	• 7.3, 7.4
	CCE	• CCE-91140-4

9.30. Ensure Software Update is Updated and Current

Make sure Software Update is updated and current.



Automatic fix can cause unplanned restarts and may lose work.

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

```
softwareupdate_date_epoch=$(/bin/date -j -f "%Y-%m-%d" "$(/usr/bin/defaults read
/Library/Preferences/com.apple.SoftwareUpdate.plist LastFullSuccessfulDate |
/usr/bin/awk '{print $1}')" "+%s")
thirty_days_epoch=$(/bin/date -v -30d "+%s")
if [[ $softwareupdate_date_epoch -lt $thirty_days_epoch ]]; then
  /bin/echo "0"
else
  /bin/echo "1"
fi
```

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

Remediation Description

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

```
/usr/sbin/softwareupdate -i -a -R
```

NOTE - This will apply to the whole system

ID	sysprefs_softwareupdate_current	
References	800-53r5	<ul style="list-style-type: none">• N/A
	CIS Benchmark	<ul style="list-style-type: none">• 1.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 7.3, 7.4
	CCE	<ul style="list-style-type: none">• CCE-91141-2

9.31. Disable SSH Server for Remote Access Sessions

SSH service *MUST* be disabled for remote access.

Remote access sessions *MUST* use FIPS validated encrypted methods to protect unauthorized individuals from gaining access.

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

```
/bin/launchctl print-disabled system | /usr/bin/grep -c '"com.openssh.sshd" => true'
```

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.openssh.sshd
```

ID	sysprefs_ssh_disable
-----------	----------------------

References	800-53r5	<ul style="list-style-type: none"> • AC-17 • CM-7, CM-7(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.4.5 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1, 4.8
	CCE	<ul style="list-style-type: none"> • CCE-91077-8

9.32. Require Administrator Password to Modify System-Wide Preferences

The system *MUST* be configured to require an administrator password in order to modify the system-wide preferences in System Preferences.

Some Preference Panes in System Preferences contain settings that affect the entire system. Requiring a password to unlock these system-wide settings reduces the risk of a non-authorized user modifying system configurations.

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

```
/usr/bin/security authorizationdb read system.preferences 2> /dev/null |
/usr/bin/grep -A 1 "<key>shared</key>" | /usr/bin/grep -c "<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/security authorizationdb read system.preferences >
/tmp/system.preferences.plist
/usr/libexec/PlistBuddy -c "Set :shared false" /tmp/system.preferences.plist
/usr/bin/security authorizationdb write system.preferences <
/tmp/system.preferences.plist
```

ID	sysprefs_system_wide_preferences_configure
-----------	--

References	800-53r5	<ul style="list-style-type: none"> • AC-6, AC-6(1), AC-6(2)
	CIS Benchmark	<ul style="list-style-type: none"> • 5.10 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 4.1
	CCE	<ul style="list-style-type: none"> • CCE-91079-4

9.33. Configure macOS to Use an Authorized Time Server

Approved time servers *MUST* be the only servers configured for use.

This rule ensures the uniformity of time stamps for information systems with multiple system clocks and systems connected over a network.

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

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

If the result is not **time-a.nist.gov,time-b.nist.gov**, 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.MCX) payload type:

```
<key>timeServer</key>
<string>time-a.nist.gov,time-b.nist.gov</string>
```

ID	sysprefs_time_server_configure
----	--------------------------------

References	800-53r5	<ul style="list-style-type: none"> • AU-12(1) • SC-45(1)
	CIS Benchmark	<ul style="list-style-type: none"> • 2.2.1 (level 1)
	CIS Controls V8	<ul style="list-style-type: none"> • 8.4
	CCE	<ul style="list-style-type: none"> • CCE-91080-2

9.34. Enable macOS Time Synchronization Daemon (timed)

The timed service *MUST* be enabled on all networked systems and configured to set time automatically from the approved time server.

This rule ensures the uniformity of time stamps for information systems with multiple system clocks and systems connected over a network.

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

```
/usr/bin/osascript -l JavaScript << EOS
$.NSUserDefaults.alloc.initWithSuiteName('com.apple.timed')\
.objectForKey('TMAutomaticTimeOnlyEnabled').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:



The following settings are in the (com.apple.ManagedClient.preferences) payload. This payload requires the additional settings to be sub-payloads within, containing their their defined payload types.

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

```
<key>TMAutomaticTimeOnlyEnabled</key>
<true/>
```

ID	sysprefs_time_server_enforce	
References	800-53r5 <ul style="list-style-type: none"> • AU-12(1) • SC-45(1) CIS Benchmark <ul style="list-style-type: none"> • 2.2.1 (level 1) CIS Controls V8 <ul style="list-style-type: none"> • 8.4 CCE <ul style="list-style-type: none"> • CCE-91081-0 	

9.35. Ensure Wake for Network Access Is Disabled

Wake for network access *MUST* be disabled.

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

```
/usr/bin/pmset -g custom | /usr/bin/awk '/womp/{print $2}'
```

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 womp 0
```

ID	sysprefs_wake_network_access_disable	
References	800-53r5 <ul style="list-style-type: none"> • N/A CIS Benchmark <ul style="list-style-type: none"> • 2.8 (level 1) CIS Controls V8 <ul style="list-style-type: none"> • 4.8 CCE <ul style="list-style-type: none"> • CCE-91146-1 	

9.36. Enable Wifi Menu

The WiFi menu *MUST* be enabled.

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

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

If the result is not **18**, 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.controlcenter) payload type:

```
<key>WiFi</key>
<integer>18</integer>
```

ID	sysprefs_wifi_menu_enable	
References	800-53r5	<ul style="list-style-type: none">• N/A
	CIS Benchmark	<ul style="list-style-type: none">• 4.2 (level 1)
	CIS Controls V8	<ul style="list-style-type: none">• 4.8, 12.6
	CCE	<ul style="list-style-type: none">• CCE-91149-5

10. Supplemental

This section provides additional information to support the guidance provided by the baselines.

10.1. CIS Manual Recommendations

List of CIS recommendations that are manual check in the CIS macOS Benchmark.

Section	Install Updates, Patches and Additional Security Software
Recommendations	1.7 Audit Computer Name

Section	System Preferences
Recommendations	2.3.3 Audit Lock Screen and Start Screen Saver Tools 2.5.1.2 Ensure all user storage APFS volumes are encrypted 2.5.1.3 Ensure all user storage CoreStorage volumes are encrypted 2.5.4 Audit Location Services Access 2.5.7 Audit Camera Privacy and Confidentiality 2.6.1.1 Audit iCloud Configuration 2.6.1.2 Audit iCloud Keychain 2.6.1.3 Audit iCloud Drive 2.6.2 Audit App Store Password Settings 2.12 Audit Automatic Actions for Optical Media 2.13 Audit Siri Settings 2.14 Audit Sidecar Settings 2.15 Audit Touch ID and Wallet & Apple Pay Settings 2.16 Audit Notification System Preference Settings 2.17 Audit Passwords System Preference Setting

Section	Logging and Auditing
Recommendations	3.7 Audit Software Inventory

Section	Network Configurations
Recommendations	4.3 Audit Network Specific Locations 4.6 Audit Wi-Fi Settings

Section	System Access, Authentication and Authorization
----------------	---

Recommendations	5.2.3 Ensure Complex Password Must Contain Alphabetic Characters Is Configured 5.2.4 Ensure Complex Password Must Contain Numeric Character Is Configured 5.2.6 Ensure Complex Password Must Contain Uppercase and Lowercase Characters Is Configured 5.5 Ensure login keychain is locked when the computer sleeps 5.15 Ensure Fast User Switching Is Disabled
------------------------	--

Section	Appendix: Additional Considerations
Recommendations	7.1 Extensible Firmware Interface (EFI) password 7.2 FileVault and Local Account Password Reset using AppleID

10.2. FileVault Supplemental

The supplemental guidance found in this section is applicable for the following rules: *
sysprefs_filevault_enforce

In macOS 11 the internal Apple File System (APFS) data volume can be protected by FileVault. The system volume is always cryptographically protected (T2 and Apple Silicon) and is a read-only volume.



FileVault uses an AES-XTS data encryption algorithm to protect full volumes of internal and external storage. Macs with a secure enclave (T2 and Apple Silicon) utilize the hardware security features of the architecture.

FileVault is described in detail here: <https://support.apple.com/guide/security/volume-encryption-with-filevault-sec4c6dc1b6e/web>.

FileVault can be enabled in two ways within the macOS. It can be managed using the `fdsetup` command or by a Configuration Profile. When enabling FileVault via either of the aforementioned methods, you will be required to enter a username and password, which must be a local Open Directory account with a valid SecureToken password.

Using the fdsetup Command

When enabling FileVault via the command line in the Terminal application, you can run the following command.

```
/usr/bin/fdsetup enable
```

Running this command will prompt you for a username and password and then enable FileVault and return the personal recovery key. There are a number of management features available when managing FileVault via the command line that are not available when using a configuration profile. More information on these management features is available in the man page for `fdsetup`.



Apple has deprecated `fdsetup` command line tool from recognizing user name and password for security reasons and may remove the ability in future versions of macOS.

Using a Configuration Profile

When managing FileVault with a configuration profile, you must deploy a profile with the payload type `com.apple.MCX.FileVault2`. When using the Enable key to enable FileVault with a configuration profile, you must include 1 of the following:

```
<key>Enable</key>
<string>On</string>
<key>Defer</key>
<true />
```

```
<key>Enable</key>
<string>On</string>
<key>UserEntersMissingInfo</key>
<true/>
```

If using the Defer key it will prompt for the user name and password at logout.

The `UserEntersMissingInfo` key will only work if installed through manual installation, and it will prompt for the username and password immediately.

When using a configuration profile, you can escrow the Recovery key to a Mobile Device Management (MDM) server. Documentation for that can be found on Apple's Developer site: <https://developer.apple.com/documentation/devicemanagement/fderecoverykeyescrow>.

It's recommended that you use a Personal Recovery key instead of an Institutional key as it will generate a specific key for each device. You can find more guidance on choosing a recover key here: https://docs.jamf.com/technical-papers/jamf-pro/administering-filevault-macos/10.7.1/Choosing_a_Recovery_Key.html.



FileVault currently only uses password-based authentication and cannot be done using a smartcard or any other type of multi-factor authentication.

10.3. Packet Filter (pf) Supplemental

The supplemental guidance found in this section is applicable for the following rules:

- `os_firewall_default_deny_require`

macOS contains an application layer firewall (ALF) and a packet filter (PF) firewall.

- The ALF can block incoming traffic on a per-application basis and prevent applications from gaining control of network ports, but it cannot be configured to block outgoing traffic.

- More information on the ALF can be found here: <https://support.apple.com/en-ca/HT201642>
- The PF firewall can manipulate virtually any packet data and is highly configurable.
 - More information on the BF firewall can be found here: <https://www.openbsd.org/faq/pf/index.html>

Below is a script that configures ALF and the PF firewall to meet the requirements defined in NIST SP 800-53 (Rev. 5). The script will make sure the application layer firewall is enabled, set logging to "detailed", set built-in signed applications to automatically receive incoming connections, and set downloaded signed applications to automatically receive incoming connections. It will then create a custom rule set and copy `com.apple.pfctl.plist` from `/System/Library/LaunchDaemons/` into the `/Library/LaunchDaemons` folder and name it `800-53.pfctl.plist`. This is done to not conflict with the system's pf ruleset.

The custom pf rules are created at `/etc/pf.anchors/800_53_pf_anchors`.

The ruleset will block connections on the following ports:

Port	Service
548	Apple File Protocol (AFP)
1900	Bonjour
79	Finger
20, 21	File Transfer Protocol (FTP)
80	HTTP
icmp	ping
143	Internet Message Access Protocol (IMAP)
993	Internet Message Access Protocol over SSL (IMAPS)
3689	Music Sharing
5353	mDNSResponder
2049	Network File System (NFS)
49152	Optical Media Sharing
110	Post Office Protocol (POP3)
995	Post Office Protocol Secure (POP3S)
631	Printer Sharing
3031	Remote Apple Events
5900	Screen Sharing
137, 138, 139, 445	Samba (SMB)
25	Simple Mail Transfer Protocol (SMTP)
22	Secure Shell (SSH)
23	Telnet

Port	Service
69	Trivial File Transfer Protocol (TFTP)
540	Unix-to-Unix Copy (UUCP)

For more on configuring the PF firewall check out the man pages on [pf.conf](#) and [pfctl](#).

```
#!/bin/bash

#enabling macos application firewall
enable_macos_application_firewall () {

    /usr/libexec/ApplicationFirewall/socketfilterfw --setglobalstate on
    /usr/libexec/ApplicationFirewall/socketfilterfw --setloggingopt detail
    /usr/libexec/ApplicationFirewall/socketfilterfw --setallowsigned on
    /usr/libexec/ApplicationFirewall/socketfilterfw --setallowsignedapp on

}

#enabling pf firewall with macsec rules
enable_pf_firewall_with_macsec_rules () {
    macsec_pfctl_plist="/Library/LaunchDaemons/macsec.pfctl.plist"

    if [[ -e "$macsec_pfctl_plist" ]]; then
        echo "LaunchDaemon already exists, flushing and reloading rules..."
        pfctl -e 2> /dev/null
        pfctl -f /etc/pf.conf 2> /dev/null
        return 0
    fi

    # copy system provided launchd for custom ruleset
    cp "/System/Library/LaunchDaemons/com.apple.pfctl.plist" "$macsec_pfctl_plist"
    #allow pf to be enabled when the job is loaded
    /usr/libexec/PlistBuddy -c "Add :ProgramArguments:1 string -e" $macsec_pfctl_plist
    #use new label to not conflict with System's pfctl
    /usr/libexec/PlistBuddy -c "Set :Label macsec.pfctl" $macsec_pfctl_plist

    # enable the firewall
    pfctl -e 2> /dev/null

    #make pf run at system startup
    launchctl enable system/macsec.pfctl
    launchctl bootstrap system $macsec_pfctl_plist

    pfctl -f /etc/pf.conf 2> /dev/null #flush the pf ruleset (reload the rules)

}

# append the macsec anchors to pf.conf
configure_pf_config_add_macsec_anchors () {
```

```

# check to see if macsec anchors exists
anchors_exist=$(grep -c '^anchor "macsec_pf_anchors"' /etc/pf.conf)

if [[ $anchors_exist == "0" ]];then
    echo 'anchor "macsec_pf_anchors"' >> /etc/pf.conf
    echo 'load anchor "macsec_pf_anchors" from
"/etc/pf.anchors/macsec_pf_anchors"' >> /etc/pf.conf
else
    echo "macsec anchors exist, continuing..."
fi
}

# Create /etc/pf.anchors/macsec_pf_anchors
create_macsec_pf_anchors () {
if [[ -e /etc/pf.anchors/macsec_pf_anchors ]]; then
    echo "macsec Anchor file exists, deleting and recreating..."
    rm -f /etc/pf.anchors/macsec_pf_anchors
fi

cat > /etc/pf.anchors/macsec_pf_anchors <<'ENDCONFIG'

anchor macsec_pf_anchors

#default deny all in, allow all out and keep state
block in all
pass out all keep state

## Allow DHCP
pass in inet proto udp from port 67 to port 68
pass in inet6 proto udp from port 547 to port 546

## Allow incoming SSH
pass in proto tcp to any port 22

#apple file service --port 548-- pf firewall rule
block in log proto tcp to any port { 548 }

#bonjour component SSDP --port 1900-- pf firewall rule
block log proto udp to any port 1900

#finger --port 79-- pf firewall rule
block log proto tcp to any port 79

#ftp --ports 20 21-- pf firewall rule
block in log proto { tcp udp } to any port { 20 21 }

#http --port 80-- pf firewall rule

```

```
block in log proto { tcp udp } to any port 80

#icmp pf firewall rule
block in log proto icmp

#imap --port 143-- pf firewall rule
block in log proto tcp to any port 143

#imaps --port 993-- pf firewall rule
block in log proto tcp to any port 993

#iTunes sharing --port 3689-- pf firewall rule
block log proto tcp to any port 3689

#mDNSResponder --port 5353-- pf firewall rule
block log proto udp to any port 5353

#nfs --port 2049-- pf firewall rule
block log proto tcp to any port 2049

#optical drive sharing --port 49152-- pf firewall rule
block log proto tcp to any port 49152

#pop3 --port 110-- pf firewall rule
block in log proto tcp to any port 110

#pop3s --port 995-- pf firewall rule
block in log proto tcp to any port 995

#remote apple events --port 3031-- pf firewall rule
block in log proto tcp to any port 3031

#screen_sharing --port 5900-- pf firewall rule
block in log proto tcp to any port 5900
#allow screen sharing from localhost while tunneled via SSH
pass in quick on lo0 proto tcp from any to any port 5900

#smb --ports 139 445 137 138-- pf firewall rule
block in log proto tcp to any port { 139 445 }
block in log proto udp to any port { 137 138 }

#smtp --port 25-- pf firewall rule
block in log proto tcp to any port 25

#telnet --port 23-- pf firewall rule
block in log proto { tcp udp } to any port 23

#tftp --port 69-- pf firewall rule
block log proto { tcp udp } to any port 69

#uucp --port 540-- pf firewall rule
```



```
block log proto tcp to any port 540
```

```
ENDCONFIG
```

```
}
```

```
####
```

```
enable_macos_application_firewall  
create_macsec_pf_anchors  
configure_pf_config_add_macsec_anchors  
enable_pf_firewall_with_macsec_rules
```

10.4. Password Policy Supplemental

The supplemental guidance found in this section is applicable for the following rules:

- `pwpolicy_lower_case_character_enforce`
- `pwpolicy_upper_case_character_enforce`
- `pwpolicy_account_inactivity_enforce`
- `pwpolicy_minimum_lifetime_enforce`

Password policies should be enforced as much as possible via Configuration Profiles. However, the following policies are currently not enforceable via Configuration Profiles, and must therefore be enabled using the `pwpolicy` command:

- Enforcing at least 1 lowercase character
- Enforcing at least 1 uppercase character
- Disabling an account after 35 days of inactivity
- Password minimum lifetime

To set the local policy to meet these requirements, save the following XML password policy to a file.

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"  
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">  
<plist version="1.0">  
  <dict>  
    <key>policyCategoryAuthentication</key>  
    <array>  
      <dict>  
        <key>policyContent</key>  
        <string>(policyAttributeFailedAuthentications &lt;  
policyAttributeMaximumFailedAuthentications) OR (policyAttributeCurrentTime &gt;  
(policyAttributeLastFailedAuthenticationTime + autoEnableInSeconds))</string>  
        <key>policyIdentifier</key>  
        <string>Authentication Lockout</string>  
        <key>policyParameters</key>
```

```

    <dict>
      <key>autoEnableInSeconds</key>
      <integer>300</integer>
      <key>policyAttributeMaximumFailedAuthentications</key>
      <integer>3</integer>
    </dict>
  </dict>
  <dict>
    <key>policyContent</key>
    <string>policyAttributeLastAuthenticationTime &gt; policyAttributeCurrentTime
- (policyAttributeInactiveDays * 24 * 60 * 60)</string>
    <key>policyIdentifier</key>
    <string>Inactive Account</string>
    <key>policyParameters</key>
    <dict>
      <key>policyAttributeInactiveDays</key>
      <integer>35</integer>
    </dict>
  </dict>
</array>
<key>policyCategoryPasswordChange</key>
<array>
  <dict>
    <key>policyContent</key>
    <string>policyAttributeCurrentTime &gt; policyAttributeLastPasswordChangeTime
+ (policyAttributeExpiresEveryNDays * 24 * 60 * 60)</string>
    <key>policyIdentifier</key>
    <string>Password Expires after 60 days</string>
    <key>policyParameters</key>
    <dict>
      <key>policyAttributeExpiresEveryNDays</key>
      <integer>60</integer>
    </dict>
  </dict>
</array>
<key>policyCategoryPasswordContent</key>
<array>
  <dict>
    <key>policyContent</key>
    <string>policyAttributePassword matches '(.*[A-Z].*){1,}+'</string>
    <key>policyIdentifier</key>
    <string>Must have at least 1 uppercase letter</string>
    <key>policyParameters</key>
    <dict>
      <key>minimumAlphaCharactersUpperCase</key>
      <integer>1</integer>
    </dict>
  </dict>
  <dict>
    <key>policyContent</key>
    <string>policyAttributeLastPasswordChangeTime &lt; policyAttributeCurrentTime

```

```

- (policyAttributeMinimumLifetimeHours * 60 * 60)</string>
  <key>policyIdentifier</key>
  <string>Minimum Password Lifetime</string>
  <key>policyParameters</key>
  <dict>
    <key>policyAttributeMinimumLifetimeHours</key>
    <integer>24</integer>
  </dict>
</dict>
<dict>
  <key>policyContent</key>
  <string>policyAttributePassword matches '.{15,}+'</string>
  <key>policyIdentifier</key>
  <string>Must be at least 15 characters</string>
  <key>policyParameters</key>
  <dict>
    <key>minimumLength</key>
    <integer>15</integer>
  </dict>
</dict>
<dict>
  <key>policyContent</key>
  <string>policyAttributePassword matches '(. *[0-9]. *){1,}+'</string>
  <key>policyIdentifier</key>
  <string>Must have at least 1 numeric value</string>
  <key>policyParameters</key>
  <dict>
    <key>minimumNumericCharacters</key>
    <integer>2</integer>
  </dict>
</dict>
<dict>
  <key>policyContent</key>
  <string>policyAttributePassword matches '(. *[a-z]. *){1,}+'</string>
  <key>policyIdentifier</key>
  <string>Must have at least 1 lowercase letter</string>
  <key>policyParameters</key>
  <dict>
    <key>minimumAlphaCharactersLowerCase</key>
    <integer>1</integer>
  </dict>
</dict>
<dict>
  <key>policyContent</key>
  <string>policyAttributePassword matches '(. *[A-Za-z]. *){1,}+'</string>
  <key>policyIdentifier</key>
  <string>Must have at least 1 Letter</string>
  <key>policyParameters</key>
  <dict>
    <key>minimumAlphaCharacters</key>
    <integer>1</integer>

```

```

    </dict>
  </dict>
  <dict>
    <key>policyContent</key>
    <string>policyAttributePassword matches '(.^[a-zA-Z0-9].*){1,}+'</string>
    <key>policyIdentifier</key>
    <string>Must have at least 1 special characters</string>
    <key>policyParameters</key>
    <dict>
      <key>minimumSymbols</key>
      <integer>1</integer>
    </dict>
  </dict>
  <dict>
    <key>policyContent</key>
    <string>none policyAttributePasswordHashes in
policyAttributePasswordHistory</string>
    <key>policyIdentifier</key>
    <string>Cannot match the last 5 passwords</string>
    <key>policyParameters</key>
    <dict>
      <key>policyAttributePasswordHistoryDepth</key>
      <integer>5</integer>
    </dict>
  </dict>
</array>
</dict>
</plist>

```

Run the following command to load the new policy file, substituting the path to the file in place of "\$pwpolicy_file".

```
/usr/bin/pwpolicy setaccountpolicies $pwpolicy_file
```



If directory services is being utilized, password policies should come from the domain.

10.5. Smartcard Supplemental

The supplemental guidance found in this section is applicable for the following rules:

- auth_ssh_password_authentication_disable
- auth_smartcard_enforce
- auth_smartcard_certificate_trust_enforce_moderate
- auth_smartcard_certificate_trust_enforce_high
- auth_smartcard_allow

- `auth_pam_sudo_smartcard_enforce`
- `auth_pam_su_smartcard_enforce`
- `auth_pam_login_smartcard_enforce`

macOS supports smartcards, such as U.S. Personal Identity Verification (PIV) cards and U.S. Department of Defense Common Access Cards (CAC). Smartcards can be used on a macOS for the following:

- Authentication (Loginwindow, Screensaver, SSH, PKINIT, Safari, Finder, and PAM Authorization (`sudo`, `login`, and `su`))
- Digital Encryption
- Digital Signing
- Remote Access (VPN:L2TP)
- Port-based Network Access Control (802.1X)
- Keychain Unlock

macOS has built-in support for USB CCID class-compliant smartcard readers.

Smartcard Pairing

The default method for using smartcards in macOS is a method called "local account pairing". Local account pairing is automatically initiated when a user inserts a smartcard into the Mac. The user is prompted to pair their smartcard with their account. If a user receives a new smartcard, the previous card must be unpaired, and the new card paired to the account. Local account pairing employs fixed key mapping with the hash of a public key on the user's smartcard with a local account.

Smartcard Attribute Mapping

Smartcards can be used to authenticate against a directory via attribute mapping configured in `/private/etc/SmartcardLogin.plist`. This file takes precedence over local account pairing. Attribute mapping matches the configured certificate field values from the smart card to the value in a directory. This may be used with network accounts, mobile accounts, or local accounts.

Smartcard Management in macOS

The following settings are available to manage smartcards (`com.apple.security.smartcard`):

Key	Type	Value
<code>userPairing</code>	bool	If false, users will not get the pairing dialog, although existing pairings will still work.
<code>allowSmartCard</code>	bool	If false, the SmartCard is disabled for logins, authorizations, and screensaver unlocking. It is still allowed for other functions, such as signing emails and web access. A restart is required for a change of setting to take effect.

Key	Type	Value
checkCertificateTrust	int	Valid values are 0-3: <ul style="list-style-type: none"> 0: certificate trust check is turned off 1: certificate trust check is turned on. Standard validity check is being performed but this does not include additional revocation checks. 2: certificate trust check is turned on, and a soft revocation check is performed. Until the certificate is explicitly rejected by CRL/OCSP, it is considered valid. This implies that unavailable/unreachable CRL/OCSP allows this check to succeed. 3: certificate trust check is turned on, plus a hard revocation check is performed. Unless CRL/OCSP explicitly states that "this certificate is OK", the certificate is considered invalid. This is the most secure value for this setting.
oneCardPerUser	bool	If true, a user can pair with only one smartcard, although existing pairings will be allowed if already set up.
enforceSmartCard	bool	If true, a user can only login or authenticate with a smartcard.
tokenRemovalAction	int	If 1, the screen saver will automatically when the smartcard is removed.
allowUnmappedUsers	int	If 1, allows users who are in a directory group to be exempt from smartcard-only enforcement. The group allowed for exemption is defined in /private/etc/SmartcardLogin.plist

A custom configuration profile (`com.apple.loginwindow`) should be created to disable automatic login when FileVault is enabled. This ensures that authorized users boot their Macs, enter a password at the pre-boot screen (which decrypts the boot volume), and are then presented with a login window where they can authenticate with a smartcard.

Key	Type	Value
DisableFDEAutoLogin	bool	If true, both Extensible Firmware Interface (EFI) login password and loginwindow PIN are required.



DisableFDEAutoLogin does not have to be set on Apple Silicon based macOS systems that are smartcard enforced as smartcards are available at pre-boot.

Trusted Authorities

The macOS allows users to specify which certificate authorities (CA) can be used for trust evaluation during smartcard authentication. Only CAs listed in the TrustedAuthorities section of the SmartcardLogin.plist will be evaluated as trusted. This setting only works if `checkCertificateTrust` is set to either 1, 2, or 3 in `com.apple.security.smartcard`.

To get the SHA-256 hash in the correct format, run the following command within terminal:

```
/usr/bin/openssl x509 -noout -fingerprint -sha256 -inform pem -in <issuer cert> |  
/usr/bin/awk -F '=' '{print $2}' | /usr/bin/sed 's://g'
```

To configure Trusted Authorities, the `SmartcardLogin.plist` should be minimally configured as below:

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"  
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">  
<plist version="1.0">  
  <dict>  
    <key>AttributeMapping</key>  
    <dict>  
      <key>fields</key>  
      <array>  
        <string>NT Principal Name</string>  
      </array>  
      <key>formatString</key>  
      <string>Kerberos:$1</string>  
      <key>dsAttributeString</key>  
      <string>dsAttrTypeStandard:AltSecurityIdentities</string>  
    </dict>  
    <key>TrustedAuthorities</key>  
    <array>  
      <string>SHA256_HASH_OF_CERTDOMAIN_1,SHA256_HASH_OF_CERTDOMAIN_2</string>  
    </array>  
  </dict>  
</plist>
```

Smartcard Enforcement Exemption

Group Exemption

Starting in macOS 10.15, enforcement on a system can be granularly configured by adding a field to `/private/etc/SmartcardLogin.plist`. The `NotEnforcedGroup` can be added to the file to list a Directory group that will not be included in smartcard enforcement. In order to activate this feature, `enforceSmartCard` and `allowUnmappedUsers` must be applied via a configuration profile (`com.apple.security.smartcard`).

To configure the `NotEnforcedGroup`, the `SmartcardLogin.plist` should be minimally configured as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>AttributeMapping</key>
  <dict>
    <key>fields</key>
    <array>
      <string>NT Principal Name</string>
    </array>
    <key>formatString</key>
    <string>Kerberos:$1</string>
    <key>dsAttributeString</key>
    <string>dsAttrTypeStandard:AltSecurityIdentities</string>
  </dict>
  <key>TrustedAuthorities</key>
  <array>
    <string>SHA256_HASH_OF_CERTDOMAIN_1,SHA256_HASH_OF_CERTDOMAIN_2</string>
  </array>
  <key>NotEnforcedGroup</key>
  <string>EXEMPTGROUP</key>
</dict>
</plist>
```

Once a system is configured for the `NotEnforcedGroup` a user can be added to the assigned group by running the following:

```
/usr/sbin/dseditgroup -o edit -a <exempt_user> -t user <notenforcegroup>
```

User Exemption

Alternatively, if a single user needs to be exempt for a period of time, `kDSNativeAttrTypePrefix:SmartCardEnforcement` can be set in the user's Open Directory record. The following values can be set:

- 0 - The system default is respected.
- 1 - Smartcard enforcement is enabled.
- 2 - Smartcard enforcement is disabled.



In Active Directory environments, the value of the `userAccountControl` attribute is respected.

Run the following command to set the exemption when booted from macOS:

```
/usr/bin/dscl . -append /Users/<username> SmartCardEnforcement 2
```


Run the following command to set the exemption when booted from Recovery:

```
/usr/bin/defaults write /Volumes/Macintosh\  
HD/var/db/dslocal/nodes/Default/users/<username> SmartCardEnforcement -array-add 2
```



When booted to recovery on an Apple Silicon Mac, run the following after setting the exemption. `/usr/sbin/diskutil apfs updatePreboot /Volumes/Macintosh\ HD`

Temporary Exemption

On an Apple Silicon Mac, if a temporary exemption is needed, `security filevault skip-sc-enforcement` will disable smartcard enforcement on next boot only.

Run the following command to set the temporary exemption when booted from Recovery:

```
/usr/bin/security filevault skip-sc-enforcement <data volume UUID> set
```

To obtain the `data volume UUID` run the following:

```
/usr/sbin/diskutil apfs listGroups | /usr/bin/awk -F: '/ Data/ { getline; gsub(/  
/, ""); print $2}'
```

Pluggable Authentication Module (PAM)

Terminal sessions in macOS can be configured for smartcard enforcement by modifying the PAM modules for `sudo`, `su`, and `login`.

```
/etc/pam.d/sudo  
# sudo: auth account password session  
auth      sufficient    pam_smartcard.so  
auth      required      pam_opendirectory.so  
auth      required      pam_deny.so  
account    required      pam_permit.so  
password   required      pam_deny.so  
session    required      pam_permit.so
```

```
/etc/pam.d/su
```

```
# su: auth account password session
```

```
auth      sufficient    pam_smartcard.so
```

```
auth      required      pam_rootok.so
```

```
auth      required      pam_group.so no_warn group=admin,wheel ruser root_only
```

```
fail_safe
```

```
account   required      pam_permit.so
```

```
account   required      pam_opendirectory.so no_check_shell
```

```
password  required      pam_opendirectory.so
```

```
session   required      pam_launchd.so
```

```
/etc/pam.d/login
```

```
# login: auth account password session
```

```
auth      sufficient    pam_smartcard.so
```

```
auth      optional      pam_krb5.so use_kcminit
```

```
auth      optional      pam_ntlm.so try_first_pass
```

```
auth      optional      pam_mount.so try_first_pass
```

```
auth      required      pam_opendirectory.so try_first_pass
```

```
auth      required      pam_deny.so
```

```
account   required      pam_nologin.so
```

```
account   required      pam_opendirectory.so
```

```
password  required      pam_opendirectory.so
```

```
session   required      pam_launchd.so
```

```
session   required      pam_uwtmp.so
```

```
session   optional      pam_mount.so
```