ninjaOne®

The Endpoint Hardening Playbook

Locking down devices with NinjaOne

Why Endpoint Hardening Is Important

At its core, endpoint (or device) hardening is the process of reinforcing security at the device level. Because securing your endpoints is fundamental to every other security action you take, the investment you make in it will have greater ROI than almost anything else. If you don't do it well, every other solution and step you take will have to be better, work harder, and have fewer gaps.

Unfortunately, many organizations still don't take the basic steps needed to support comprehensive endpoint hardening. According to IBM some of the top IT security challenges in the last year include:

- Cloud computing
- Remote, hybrid, BYOD workforces
- Proliferation of endpoints
- Expansion of IoT
- Rise of AI

Some key issues that negatively impact an organization's cyber resiliency include:

- Lack of usable backups
- Unmanaged legacy systems
- Limited use of EDR
- Lack of patch & vulnerability management
- Inadequate admin password controls
- Overly complex MFA

Recommended actions to boost cyber resiliency include:

- Take regular backups
- Patch OS and 3rd party applications
- Configure and control application use
- Encrypt drives and harden endpoint configurations
- Restrict user privileges
- Enforce complex passwords and MFA
- Enable AV / NGAV / antimalware solutions
- Secure portable media
- Enable the firewall on your devices

Basic security hygiene prevents 98% cyber attacks

What Does Device Hardening Encompass?

Device hardening includes any changes you make to a device that helps improve its security.

Account access protection

- Enable and enforce MFA
- Remove extraneous accounts
- Change default admin accounts
- Enforce least privilege access across user accounts
- Block end users from installing apps
- Enforce strong passwords

Device configuration

- Enable secure boot
- Disable USB
- Encrypt disk
- Block net calls from applications (notepad, wscript, cscript, etc.)
- Reduce port exposure
- Enable and expand logging
- Disable insecure protocols like SMBv1, Telnet, and HTTP
- Password protect BIOS/UEFI

Software management

- Remove potentially malicious apps
- Remove unsupported software
- Deploy antivirus / EDR
- Deploy password management solutions
- Enable firewall
- Remove old executables
- Prevent end users from installing apps

Auditing

- Audit device hardening

Note that this is not an exhaustive list, but it does offer a starting point for organizations looking for next steps in their endpoint hardening process. Not every device hardening activity will be applicable to every environment and many will need to be adapted to your own environment. When improving endpoint security, remember that baselines are constantly changing, so security approaches should always be evaluated and refreshed on regularly.

It's also important to note that there are a number of critical actions that you may take to bolster your organization's security, but are not included in endpoint hardening, including:

- Identity and access management
- Advanced security solutions (SIEM, advanced AV, etc.)
- Security awareness training for end users
- Network strategy
- Cloud application security
- Mobile threat defense

While all of these actions are crucial to ensuring security, they don't specifically target security at the device level.

Automate Endpoint Hardening

In general, IT automation:

- Decreases the potential for human error
- Reduces the number of manual tasks
- Lowers costs
- Standardizes device management and service delivery
- Improves IT employee satisfaction
- Enhances the end-user experience
- Helps to support compliance

As you can see, automation can make the endpoint hardening process much simpler, more efficient, and cheaper in the long run. Additionally, since processes are set to run automatically, organizations can more quickly limit exposure to any potential vulnerabilities. The less time a device is exposed, the more secure it will be.

NinjaOne offers a number of tools that help organizations easily implement automated IT workflows. In the next section, we'll demonstrate five examples of how organizations can use NinjaOne's automation tools to improve device security, including:

1. Scheduled scripts and tasks
Implement these when you want to act against devices in a policy at a specific time or times.

2. Script result conditions
Use these conditions to regularly check information on a device and act based on the returned results.

3. Condition-triggered script
Set these up to respond immediately to a state change on a device

4. Custom field-triggered scripts
Create these when you need information NinjaOne doesn't collect by default or for multi-step, complex automations

Adding a custom script to NinjaOne

Because so many automations need to be customized to individual environments, custom scripts are critical to NinjaOne's automation. Therefore, it's important to know how to add new scripts to your script library within the NinjaOne platform.

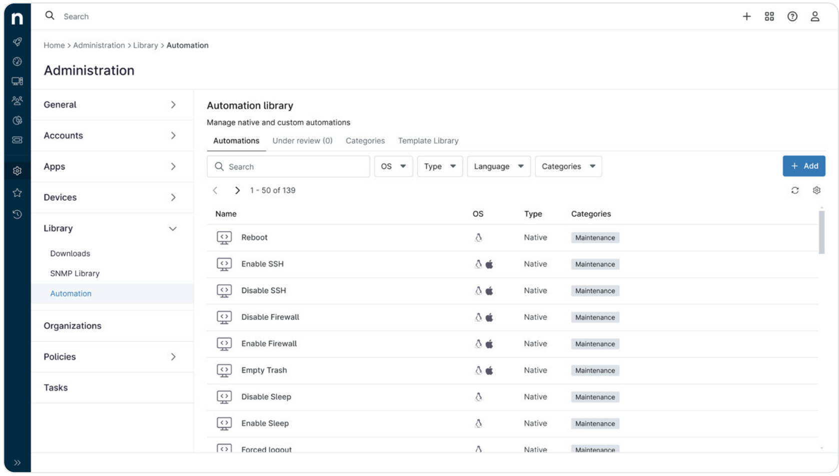
First, in the NinjaOne dashboard and navigate to 'Administration' on the left-hand side. Click on 'Library' and 'Scripting' to access your script library. (Image 1)

To add a new script, you can either:

+ Add a new script using the script editor. (Image 2)
+ Import a new script using the template library
+ Import a new script from your computer

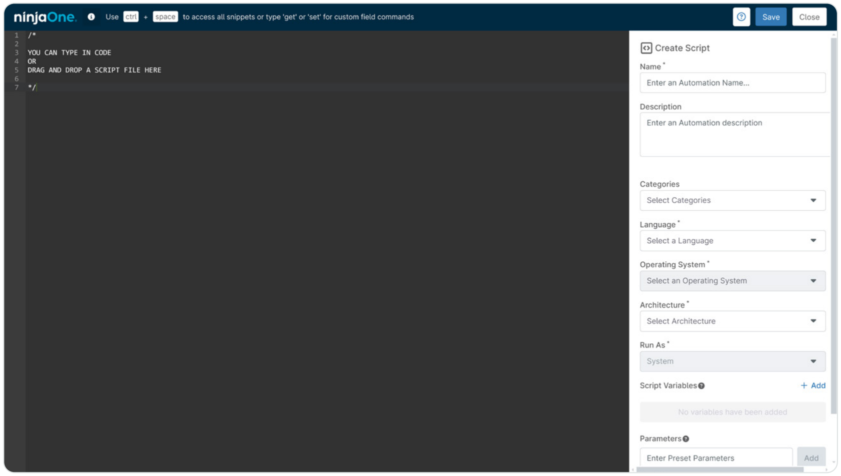
For help with custom scripts, you can visit the NinjaOne Dojo, which is our NinjaOne customer community. It's full of scripts uploaded by fellow NinjaOne users.

Note: Thoroughly test any scripts from the Dojo before rolling out. Though the NinjaOne team does keep an eye on uploaded community scripts, they are not officially released by NinjaOne.



Name	OS	Type	Categories
Reboot	🐧	Native	Maintenance
Enable SSH	🐧 🍎	Native	Maintenance
Disable SSH	🐧 🍎	Native	Maintenance
Disable Firewall	🐧 🍎	Native	Maintenance
Enable Firewall	🐧 🍎	Native	Maintenance
Empty Trash	🐧 🍎	Native	Maintenance
Disable Sleep	🐧	Native	Maintenance
Enable Sleep	🐧	Native	Maintenance
Forced logout	🐧	Native	Maintenance

Image 1. Script library



```
/*

YOU CAN TYPE IN CODE
OR
DRAG AND DROP A SCRIPT FILE HERE

*/
```

Image 2. Script editor

Five Ways to Automate Device Hardening

How to use the NinjaOne console to automate and support endpoint hardening.

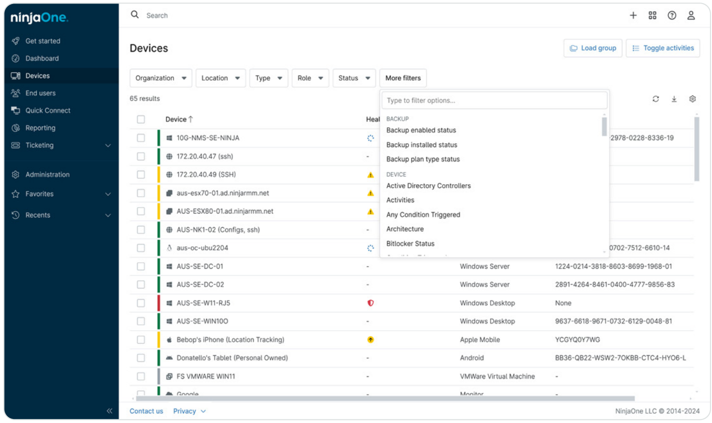
1. Deploy device security configurations

This example uses the Scheduled Tasks feature to automate device hardening outside of policies.

When setting up new devices, you can use tools that are automatically built into recent versions of Windows to improve device security. In this example, we're using Bitlocker, which is included on every Windows 10 and 11 workstation. With NinjaOne, you can track the status of Bitlocker, find devices that have Bitlocker disabled, and re-enable it using a custom script.

Note: If you don't already have the script to enable Bitlocker added to your console, follow the steps in the previous section to add the script to your library. For this demonstration, we used a community PowerShell script. We recommend naming it 'Enable Bitlocker' to easily find it during the scheduling process.)

In the NinjaOne console, navigate to 'Search' on the left-hand side and use the 'Additional Filters +' option to sort by Bitlocker Status. (images 3 and 4)



Devices

65 results

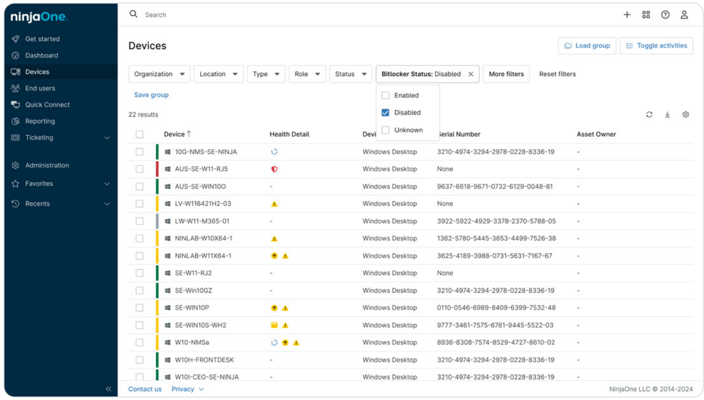
Type to filter options...

BACKUP
- Backup enabled status
- Backup installed status
- Backup plan type status

DEVICE
- Active Directory Controllers
- Activities
- Any Condition Triggered
- Architecture
- Bitlocker Status

Device ↑	Heal		
10G-NMS-SE-NINJA			2978-0228-8336-19
172.20.40.47 (ssh)	-		
172.20.40.49 (SSH)	⚠		
aus-esx70-01.ad.ninjarmm.net	⚠		
AUS-ESX80-01.ad.ninjarmm.net	⚠		
AUS-NK1-02 (Configs, ssh)	-		
aus-oc-ubu2204			0702-7512-6610-14
AUS-SE-DC-01	-	Windows Server	1224-0214-3818-8603-8699-1968-01
AUS-SE-DC-02	-	Windows Server	2891-4264-8461-0400-4777-9856-83
AUS-SE-W11-RJ5		Windows Desktop	None
AUS-SE-WIN10O	-	Windows Desktop	9637-6618-9671-0732-6129-0048-81
Bebop's iPhone (Location Tracking)		Apple Mobile	YCGYQ0Y7WG
Donatello's Tablet (Personal Owned)	-	Android	BB36-QB22-WSW2-7OKBB-CTC4-HYO6-L
FS VMWARE WIN11	-	VMWare Virtual Machine	-
Google	-	Monitor	-

Image 3. Search



Devices

Bitlocker Status: Disabled

□ Enabled
☑ Disabled
□ Unknown

22 results

Device ↑	Health Detail	Devi	Serial Number	Asset Owner
10G-NMS-SE-NINJA		Windows Desktop	3210-4974-3294-2978-0228-8336-19	-
AUS-SE-W11-RJ5		Windows Desktop	None	-
AUS-SE-WIN10O	-	Windows Desktop	9637-6618-9671-0732-6129-0048-81	-
LV-W116421H2-03	⚠	Windows Desktop	None	-
LW-W11-M365-01	-	Windows Desktop	3922-5922-4929-3378-2370-5788-05	-
NINLAB-W10X64-1	⚠	Windows Desktop	1362-5780-5445-3653-4499-7526-38	-
NINLAB-W11X64-1	⚠	Windows Desktop	3625-4189-3988-0731-5631-7167-67	-
SE-W11-RJ2	-	Windows Desktop	None	-
SE-Win10GZ	-	Windows Desktop	3210-4974-3294-2978-0228-8336-19	-
SE-WIN10P	⚠	Windows Desktop	0110-0546-6989-8409-6399-7532-48	-
SE-WIN10S-WH2	⚠	Windows Desktop	9777-3461-7575-6761-9445-5522-03	-
W10-NMSa	⚠	Windows Desktop	8936-8308-7574-8529-4727-8610-02	-
W10H-FRONTDESK	-	Windows Desktop	3210-4974-3294-2978-0228-8336-19	-
W10I-CEO-SE-NINJA	-	Windows Desktop	3210-4974-3294-2978-0228-8336-19	-

Image 4. 'Bitlocker Disabled' status filter

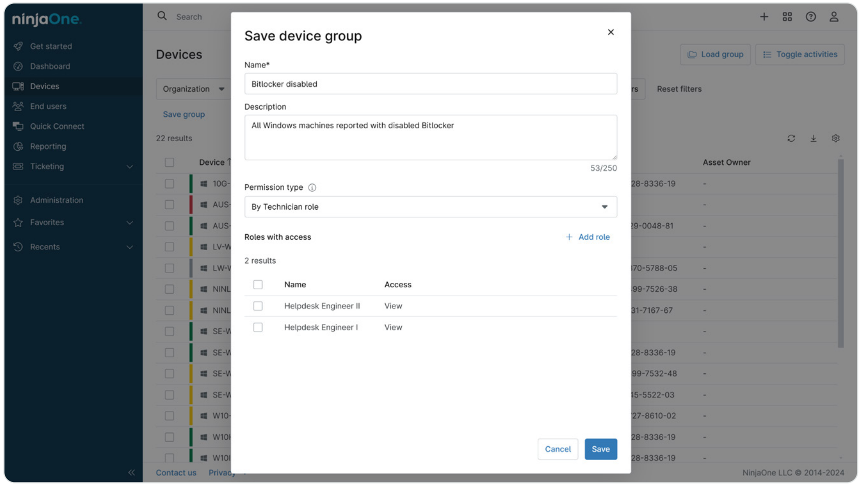
1. Deploy device security configurations on device setup *(continued)*

You can also use the dropdowns to the left of the additional filters option to filter by organization, location, device type, role, and status. After filtering your desired endpoints, you can create a dynamic group with those that have Bitlocker disabled.

To create a dynamic group of these devices, you'll simply click on 'Save' above the filtering options and choose a name for your group. (Image 5)

Once your dynamic group is created, it will always show you up-to-date information. As you enable Bitlocker on these devices, it will fall out of this group. As you onboard new machines that don't have Bitlocker enabled, they'll show up in this group.

It's a good idea to add some automated remediation to this process. To do that, navigate to 'Administration' on the left-hand sidebar and go down to 'Tasks' to add a New Task. The 'New Task' button will be on the top right-hand side of the Tasks page. (Image 6)



Save device group

Name*: Bitlocker disabled

Description: All Windows machines reported with disabled Bitlocker

53/250

Permission type: By Technician role

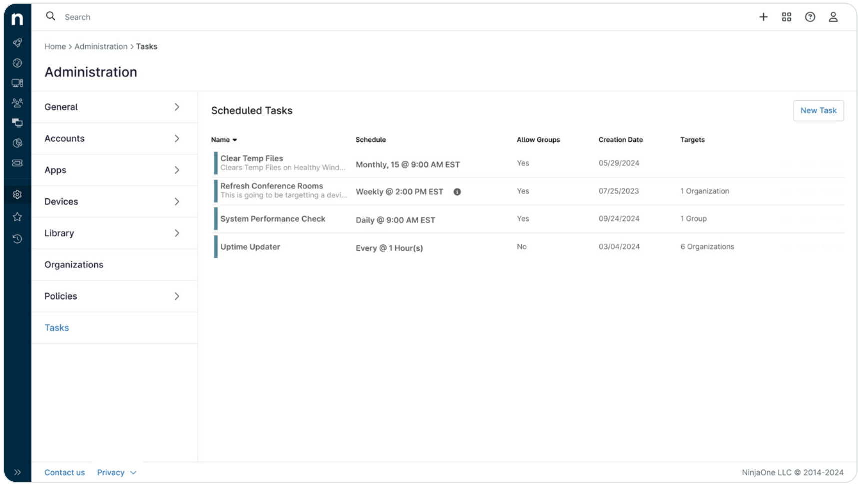
Roles with access — + Add role

2 results

	Name	Access
□	Helpdesk Engineer II	View
□	Helpdesk Engineer I	View

Cancel | Save

Image 5. Dynamic group with 'Bitlocker Disabled' devices



Scheduled Tasks

Name	Schedule	Allow Groups	Creation Date	Targets
Clear Temp Files Clears Temp Files on Healthy Wind...	Monthly, 15 @ 9:00 AM EST	Yes	05/29/2024	
Refresh Conference Rooms This is going to be targetting a devi...	Weekly @ 2:00 PM EST	Yes	07/25/2023	1 Organization
System Performance Check	Daily @ 9:00 AM EST	Yes	09/24/2024	1 Group
Uptime Updater	Every @ 1 Hour(s)	No	03/04/2024	6 Organizations

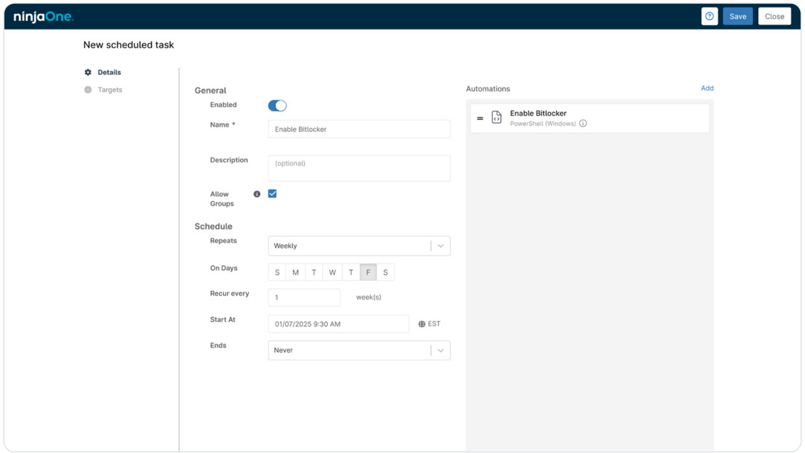
Image 6. Scheduled tasks

After creating a new scheduled task, add a name, your desired schedule (in this example, it's every Friday at 6:00 p.m. CST), and a script on the right-hand side. (Image 7)

Once you have the 'Enable Bitlocker' script added to your library, you be able to search for it. Image 8 shows what the script will look like.

After you've selected the task schedule set and script, navigate to 'Targets' and add a new target on the right-hand side. You can choose from organization, device, or group. In this example, we selected Group and search for the 'Bitlocker Disabled' dynamic group that was created earlier. (Image 9)

Other examples of using dynamic groups with scheduled scripts include disabling mass storage devices, setting UAC, and more. Within the NinjaOne Template Library in the console, you'll find a number of pre-designed script templates.



New scheduled task

General

Enabled: ●

Name *: Enable Bitlocker

Description: (optional)

Allow Groups: ☑

Schedule

Repeats: Weekly

On Days: S M T W T F S

Recur every: 1 week(s)

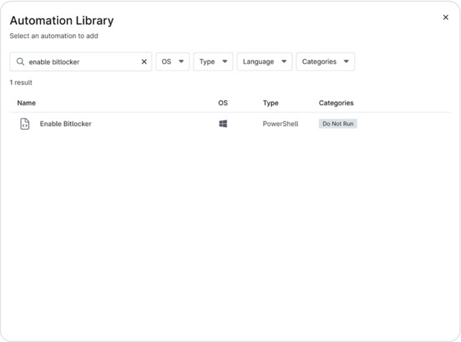
Start At: 01/07/2025 9:30 AM EST

Ends: Never

Automations

Enable Bitlocker
PowerShell (Windows)

Image 7. Creating a new scheduled task



Automation Library

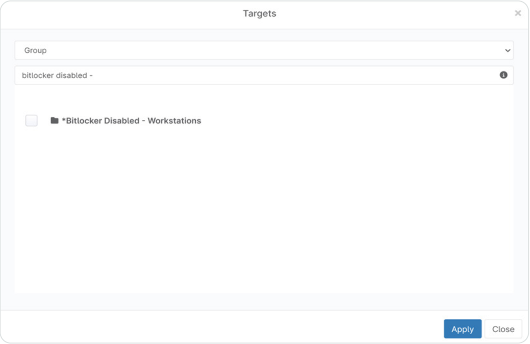
Select an automation to add

enable bitlocker

1 result

Name	OS	Type	Categories
Enable Bitlocker		PowerShell	Do Not Run

Image 8. Script list



Targets

Group

bitlocker disabled -

□ *Bitlocker Disabled - Workstations

Image 9. 'Bitlocker Disabled' dynamic group target

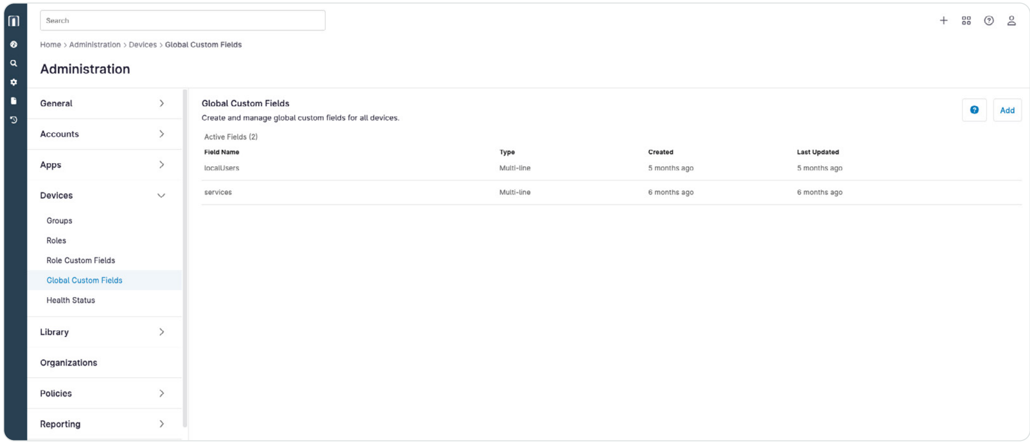
2. Enabling device firewall and blocking outbound net connections

To detect a device state and trigger an automation, use custom fields and policy conditions.

To check on the status of the device firewall in the NinjaOne console, focus on the custom field and scheduled scripts mechanisms. Custom fields can be used in a variety of ways, but for this example, we'll demonstrate how to store the output of a PowerShell script.

To add a new custom field, navigate to the 'Administration' tab on the left-hand side of your dashboard and open the 'Devices' section. Click on 'Global Custom Fields' and add a new custom field at the top right-hand side. (Image 10)

Create a new multi-line custom field, called 'Firewall Status.' You'll see a new script box similar to the one shown in Image 11. Set the script permissions to 'Read/Write.' In most cases, you would also want the Technician field set to 'Read Only.' Once you've done that, click the 'Save' button.



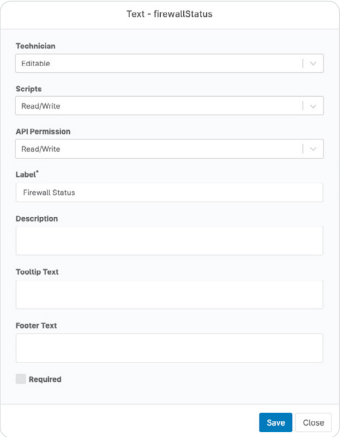
Global Custom Fields

Create and manage global custom fields for all devices.

Active Fields (2)

Field Name	Type	Created	Last Updated
localUsers	Multi-line	5 months ago	5 months ago
services	Multi-line	6 months ago	6 months ago

Image 10. Global custom fields



Text - firewallStatus

Technician: Editable

Scripts: Read/Write

API Permission: Read/Write

Label*: Firewall Status

Description:

Tooltip Text:

Footer Text:

□ Required

Image 11. New global custom field

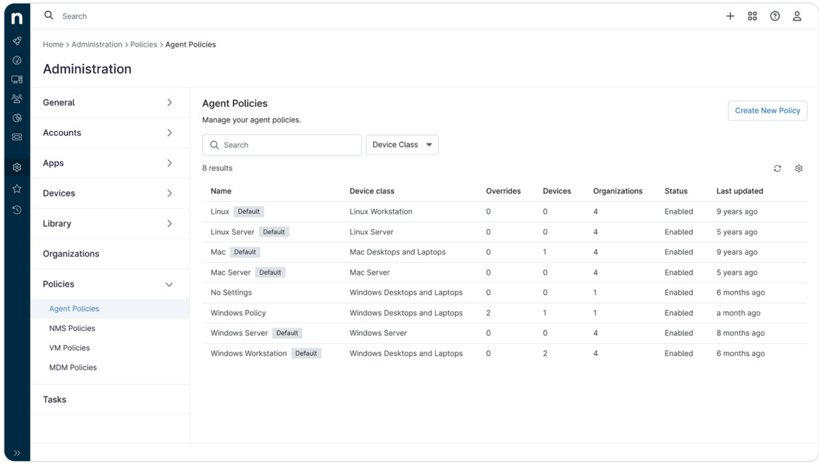
2. Enabling device firewall and blocking outbound net connections *(continued)*

Once you've created the 'Firewall Status' custom field, navigate to your policies and choose the policy you'd like to manage. (Note: If you're unfamiliar with policy setup, check out the NinjaOne policy efficiency webinar or Dojo KB article.) (Image 12)

On the policy page, go to 'Scheduled Automations' and add a new scheduled automation using the button above the list of scheduled automations. (Image 13)

Once you have your script named and set a regular schedule, add the 'Firewall - Audit Status' automation from your library on the right-hand side. You'll have options to decide which network profiles you want to monitor and a data field to enter the name of the custom field created on the previous page.

Click Save and this will add the scheduled automation to your policy. Next, go to the 'Conditions' menu and add a new condition. In this new condition, click on 'Select a condition' at the very top and choose 'Custom Fields' from the dropdown.



Home > Administration > Policies > Agent Policies

Administration

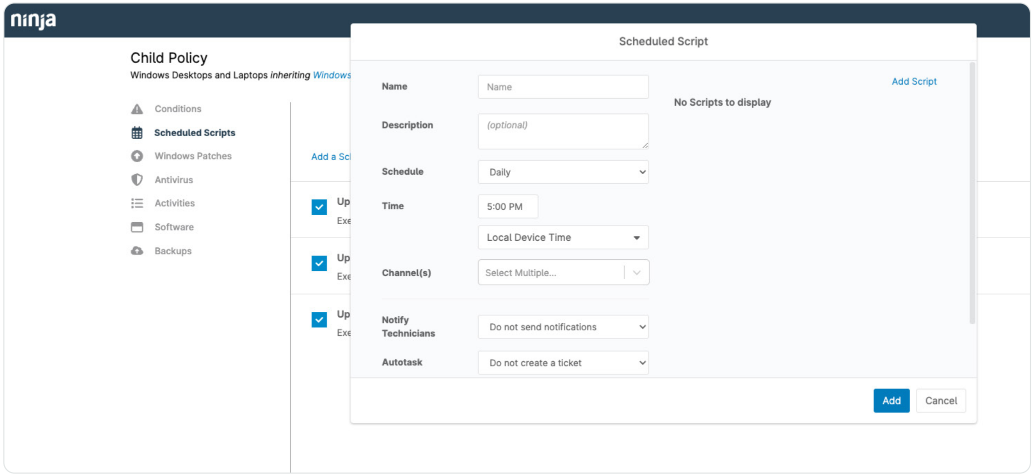
Agent Policies

Manage your agent policies.

8 results

Name	Device class	Overrides	Devices	Organizations	Status	Last updated
Linux Default	Linux Workstation	0	0	4	Enabled	9 years ago
Linux Server Default	Linux Server	0	0	4	Enabled	5 years ago
Mac Default	Mac Desktops and Laptops	0	1	4	Enabled	9 years ago
Mac Server Default	Mac Server	0	0	4	Enabled	5 years ago
No Settings	Windows Desktops and Laptops	0	0	1	Enabled	6 months ago
Windows Policy	Windows Desktops and Laptops	2	1	1	Enabled	a month ago
Windows Server Default	Windows Server	0	0	4	Enabled	8 months ago
Windows Workstation Default	Windows Desktops and Laptops	0	2	4	Enabled	6 months ago

Image 12. Agent policies



Child Policy

Windows Desktops and Laptops *inheriting* Windows

Scheduled Script

Name: Name

Description: (optional)

Schedule: Daily

Time: 5:00 PM

Local Device Time

Channel(s): Select Multiple...

Notify Technicians: Do not send notifications

Autotask: Do not create a ticket

No Scripts to display

Image 13. New scheduled script

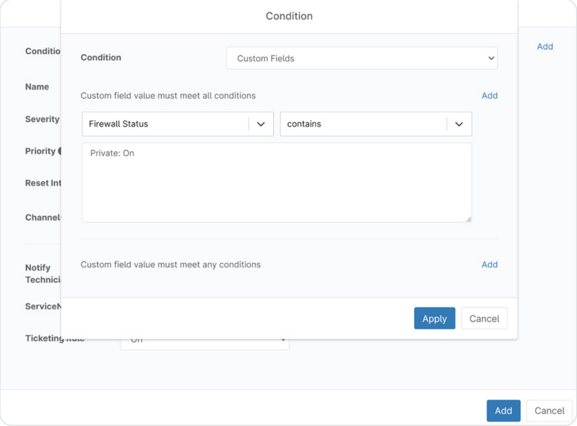
2. Enabling device firewall and blocking outbound net connections *(continued)*

After choosing custom fields, click ‘Add’ next to ‘Custom field value must meet all conditions’. (Image 14)

In the first dropdown, you’ll select the Firewall Status custom field you created. In the second dropdown, you’ll select ‘contains’ from the list. Below those two dropdowns, you’ll type ‘Off’ in the text box (meaning that the firewall is disabled). (Image 15)

Once added, you should be able to now add a script on the right-hand side. (Image 16)

Find the ‘Enable or Disable Windows Firewall’ that was previously imported, and select it. There will be an option to either Enable or Disable the firewall, and you can optionally check the box to block all inbound connections.



Condition: Custom Fields

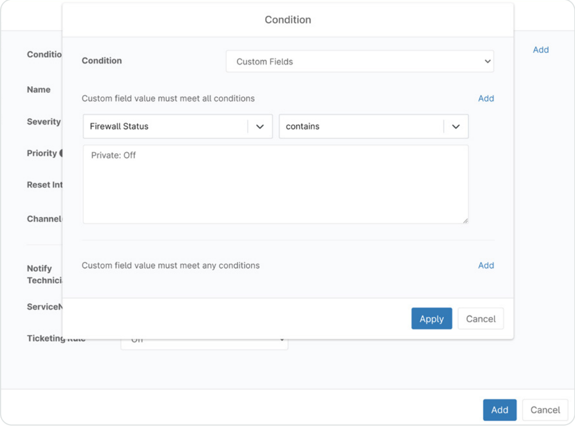
Custom field value must meet all conditions

Firewall Status | contains

Private: On

Custom field value must meet any conditions

Image 14. Custom field conditions



Condition: Custom Fields

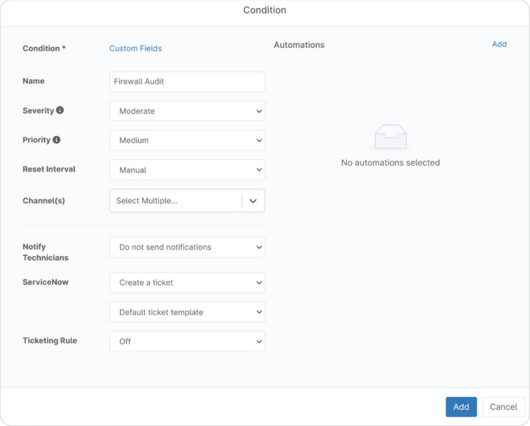
Custom field value must meet all conditions

Firewall Status | contains

Private: Off

Custom field value must meet any conditions

Image 15. Condition parameters



Condition *: Custom Fields

Name: Firewall Audit

Severity: Moderate

Priority: Medium

Reset Interval: Manual

Channel(s): Select Multiple...

Notify Technicians: Do not send notifications

ServiceNow: Create a ticket

Default ticket template

Ticketing Rule: Off

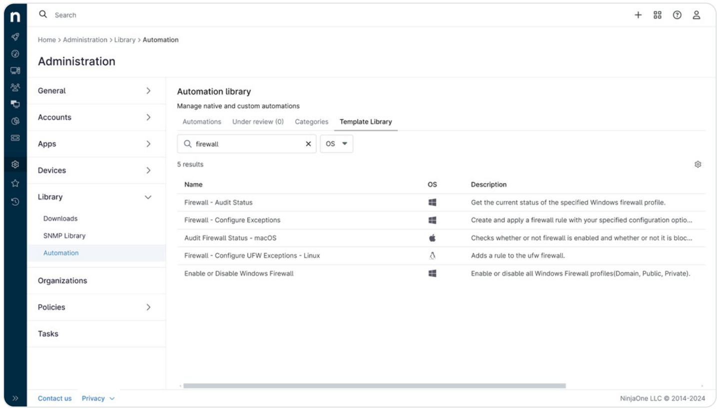
Image 16. Condition details

2. Enabling device firewall and blocking outbound net connections *(continued)*

If you haven't yet added the Enable or Disable Windows Firewall script to your Script Library, navigate to the Administration page, go to the 'Library' section and go to 'Scripting' to find your Script Library. The Enable or Disable Windows Firewall script is built in to the NinjaOne platform, so you can go to the Template Library and import it. (Image 17)

Once imported, it will appear in the dropdown to add as a script, and you can apply the condition to your policy.

To enhance firewall protection, you can also add a custom script to block outbound network communications. To add that script, follow the same custom script instructions shown on Page 5.



Automation library

Manage native and custom automations

Automations | Under review (0) | Categories | Template Library

firewall | OS

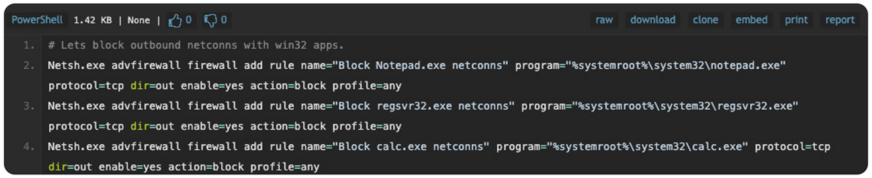
5 results

Name	OS	Description
Firewall - Audit Status	⊞	Get the current status of the specified Windows firewall profile.
Firewall - Configure Exceptions	⊞	Create and apply a firewall rule with your specified configuration optio...
Audit Firewall Status - macOS		Checks whether or not firewall is enabled and whether or not it is bloc...
Firewall - Configure UFW Exceptions - Linux		Adds a rule to the ufw firewall.
Enable or Disable Windows Firewall	⊞	Enable or disable all Windows Firewall profiles(Domain, Public, Private).

Image 17. Set-WindowsFirewall script in template library

For this example, we used a Block Outbound NetConns for win32 apps PowerShell script. (Note: this is not an official NinjaOne script, so please test carefully.) (Image 18)

Within that custom script, add the desired applications for which you'd like to block internet access. For example, it's unlikely that Windows calculator or Notepad will need internet access (but can be faked and used as vectors of attack), so you can add them to the list of any applications within the custom script itself. Once added to your Script Library, you can add the Block Outbound NetConns script to the same Check Firewall condition that you added the Block Outbound NetConns for win32 apps script.



```powershell
# Lets block outbound netconns with win32 apps.
Netsh.exe advfirewall firewall add rule name="Block Notepad.exe netconns" program="%systemroot%\system32\notepad.exe" protocol=tcp dir=out enable=yes action=block profile=any
Netsh.exe advfirewall firewall add rule name="Block regsvr32.exe netconns" program="%systemroot%\system32\regsvr32.exe" protocol=tcp dir=out enable=yes action=block profile=any
Netsh.exe advfirewall firewall add rule name="Block calc.exe netconns" program="%systemroot%\system32\calc.exe" protocol=tcp dir=out enable=yes action=block profile=any
```

Image 18. Block Outbound NetConns for win32 apps PowerShell Script

3. Enabling, expanding, and parsing logs

This example uses custom fields and policy to trigger automation on a device setup.

Logs help track device configuration and events happening on that device. It's important to review your logs to determine if they need to be expanded in order to provide an accurate view into the health and security of your IT environment.

Before taking additional steps, add a new custom script to your library specifically for expanding your event logs. Navigate to Administration -> Library -> Automation, then click on the template library. Search for 'Set Event Log Max Size' and import that automation into your library.

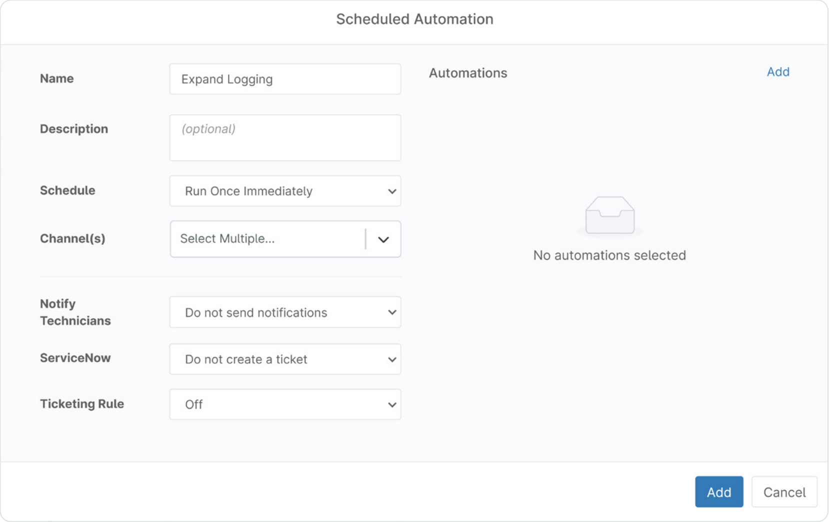
Navigate to your chosen policy page and click on 'Scheduled Automations.' This scheduled script uses the 'Run Once Immediately' cadence, running on all of the devices within your chosen policy.

The scheduling option 'Run Once Immediately' runs when devices are online, runs on any offline devices once they're back online, and runs on any new devices that join this policy. (Image 19)

Add your 'Set Event Log Max Size' custom script on the right-hand side. There are data fields to enter the sources that should be expanded – for example you can enter 'Security,System,Application', and the maximum size of the logs. From there, you can apply this scheduled script to run immediately.

In addition to expanding logs, monitoring your event viewer from privilege escalation is another way to add a layer of endpoint security. This process will take place in the 'Conditions' tab where you can set up a new condition.

Click on the "Use template" option and select the 'Security' group on the left side. Select from some of the events that can be monitored. For example, 'Member Added to Security Group' would notify us of privilege escalation. While it is possible to add remediation steps to automatically disable users until the situation is investigated, this guide does not contain steps to set up remediation.



Scheduled Automation

Name	Expand Logging
Description	(optional)
Schedule	Run Once Immediately
Channel(s)	Select Multiple...
Notify Technicians	Do not send notifications
ServiceNow	Do not create a ticket
Ticketing Rule	Off

Automations

No automations selected

Image 19. Expand logging scheduled scripts

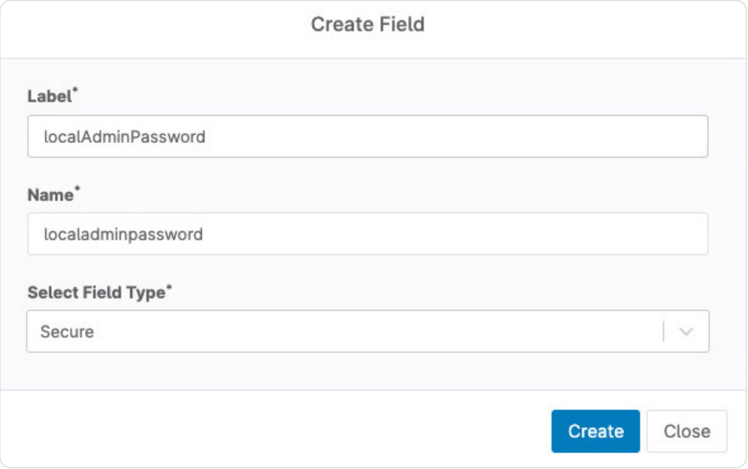
4. Creating a local admin account and automating password rotation

This example describes how to create automations for new device setup and admin password changes.

NinjaOne can be used to rotate passwords for local accounts, storing those passwords in secure custom fields that require MFA in order to be viewed.

To create the secure field, navigate to Administration -> Devices -> Global Custom Fields, and create a new custom field using the 'Secure' type. Use the label 'Local Admin Password', which should result in the name field auto-populating with 'localAdminPassword'. (Image 20)

Navigate to Administration -> Library -> Automation, and add a new script. Import this script into your environment. (Note: This is not an official NinjaOne script, please test in your environment thoroughly before using!)



Create Field

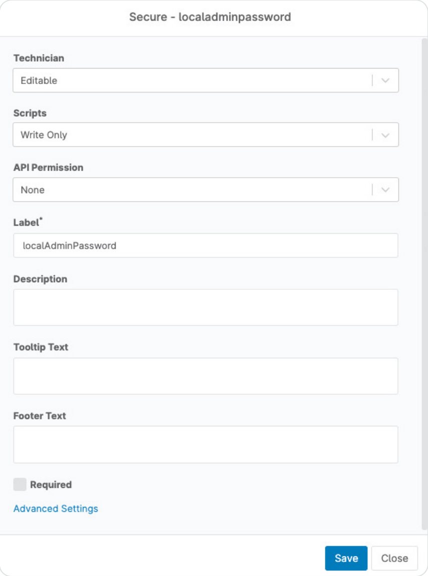
Label*: localAdminPassword

Name*: localadminpassword

Select Field Type*: Secure

Image 20. localAdminPassword field

A Secure field is specifically built for securely handling credentials. It is not visible in plain text, requires MFA to view, and is fully encrypted. Auditing is available so you can view who has access to the Secure field that's been added. Once you've clicked on 'Create,' you'll see a new box with some dropdown options. For this field, set your Scripts to 'Write Only' and ensure the field is 'Read Only' by technicians because this script creates a service account, generates a random alphanumeric character string as the password, and adds it as a Secure custom field. Once the password is generated, the script will write the password back into the Secure field. (Image 21)



Secure - localadminpassword

Technician: Editable

Scripts: Write Only

API Permission: None

Label*: localAdminPassword

Description:

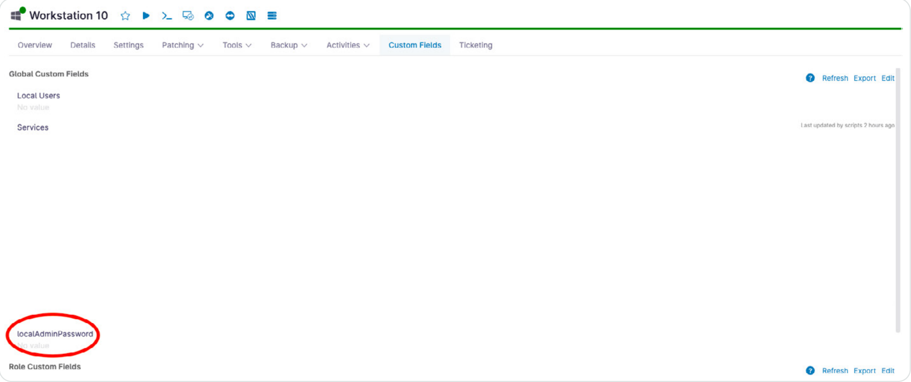
Tooltip Text:

Footer Text:

□ Required

Image 21. Secure custom field

This means that you don't have standardized passwords across the board and can go into the individual device to see the localAdminPassword in the Custom Fields section of the device page. (Image 22)



Workstation 10

Global Custom Fields

Local Users
No value

Services

Last updated by scripts 2 hours ago

localAdminPassword
No value

Role Custom Fields

Image 22. Admin password on device page

Configure a scheduled task or scheduled automation in policy to run the script on a regular basis – each time the script is executed, the specified local administrator will be created, and if already created, the password will be randomized and written into the secure custom field.

5. Detecting and removing potentially malicious software

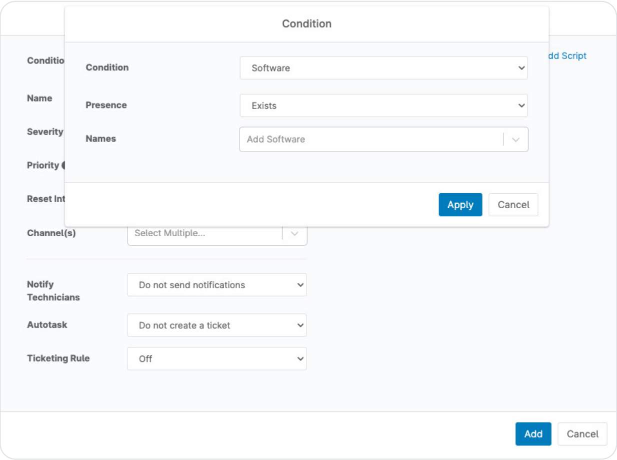
This example shows how to use a policy condition to detect a state change, such as 'software installed,' and trigger an automation to remediate the issue.

You can use NinjaOne's automation tools to detect and remove malicious or unwanted software easily from endpoints. Before you create the software detection condition, you'll need to import the 'Uninstall a Windows Application' from the NinjaOne template library using the instructions on Page 13.

Once the template library script is imported, go back into the policy that you want to update and go to the Conditions tab, and add a new condition. After choosing 'Select a condition' from the top choose 'Software' from the first dropdown and choose 'Exists' from the second dropdown. (Image 23)

After selecting both of those, enter the name of the software that you'd like to detect. If you add asterisks around the software name, it will pull in anything that uses that name in the software title, not just exact matches. Once you've saved that information, you can add the script on the right-hand side. (Image 24)

To ensure you're fully remediating your issues, test and confirm that this PowerShell script or uninstaller will successfully remove the application before deploying fully.



Condition

Condition: Software

Presence: Exists

Names: Add Software

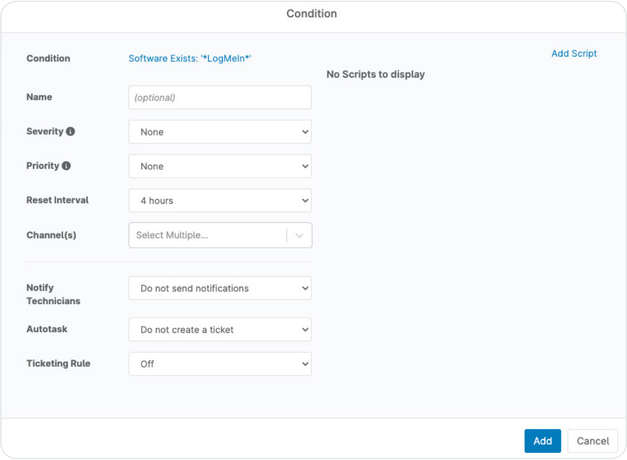
Channel(s): Select Multiple...

Notify Technicians: Do not send notifications

Autotask: Do not create a ticket

Ticketing Rule: Off

Image 23. Software exists condition



Condition

Condition: Software Exists: '*LogMeIn*'

No Scripts to display

Name: (optional)

Severity: None

Priority: None

Reset Interval: 4 hours

Channel(s): Select Multiple...

Notify Technicians: Do not send notifications

Autotask: Do not create a ticket

Ticketing Rule: Off

Image 24. Condition details

Resources

Endpoint hardening is essential. With proper automation, it can be easy to implement and maintain. If you've been looking for a tool to help you automate your IT workflow, try NinjaOne for free

Try NinjaOne for free

The following documents can help you when securing your network

Top 5 IT Security Fundamentals
OMDIA Leading IT Trends
Endpoint Hardening Checklist

About NinjaOne

NinjaOne automates the hardest parts of IT, delivering visibility, security, and control over all endpoints for more than 20,000 customers.

The NinjaOne automated endpoint management platform is proven to increase productivity, reduce security risk, and lower costs for IT teams and managed service providers. NinjaOne is obsessed with customer success and provides free and unlimited onboarding, training, and support.

NinjaOne is #1 on G2 in endpoint management, patch management, remote monitoring and management, and mobile device management.

ninjaOne®