deviceTRUST 23.1

Contents

Welcome

September 6, 2025

The deviceTRUST® documentation provides information about the installation, setup and a product reference for deviceTRUST.

Quick setup

To get started with deviceTRUST, choose your Architecture and then take a look at the Getting Started guide.

More information

- Architecture describes the different deployment scenarios.
- Getting Started provides the essential steps for a successful deviceTRUST installation.
- Download provides links that can be used to download the latest deviceTRUST Software.
- Installation details some important usage scenarios, plus how to install the deviceTRUST Console, Agent and Client Extension.
- Templates details the templates which can be used to quickly implement a use case.
- Reporting contains information about analyzing and reporting contextual information of your users and devices.
- Reference provides a set of reference material describing the more advanced features of deviceTRUST.
- Troubleshooting important steps are described to help troubleshoot the deviceTRUST installation and configuration.
- Knowledge Base provides a useful resource for common problems and resolutions.
- Releases contains the release notes detailing new features and bug fixes within the released deviceTRUST products.

Architecture

July 23, 2025

- Local Scenario
- Remote Scenario
- OS Compatibility

- Platform Compatibility

Local Scenario

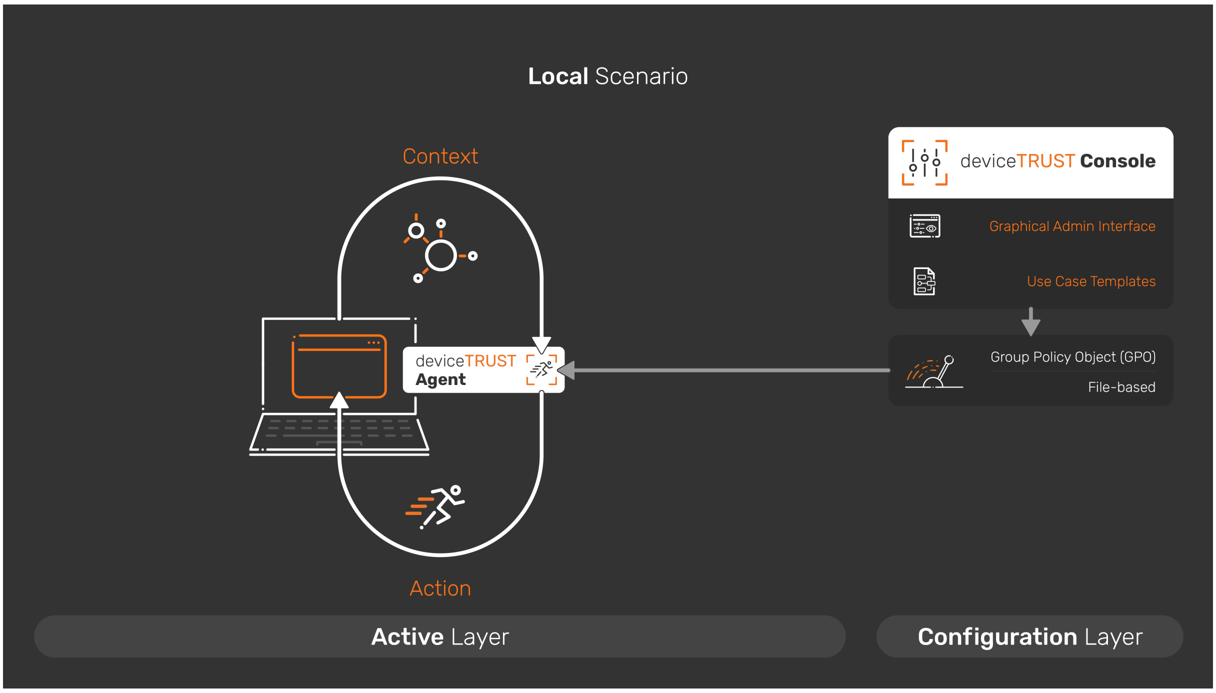
deviceTRUST requires only one main component when installing on local devices, the deviceTRUST Agent. The deviceTRUST component can be installed and configured within minutes and can be fully integrated with existing deployment processes and management tools. No additional infrastructure (e.g. a database or a web server) is required for deviceTRUST to be installed in your environment.

deviceTRUST® Agent

This component needs to be installed on the local device. The Property Reference describes which properties of the local agent are available within the users' session.

Architecture - Microsoft Windows local devices

The following diagram details the deviceTRUST architecture when the agent is installed on a Windows OS, with deviceTRUST making the user and device context information available within the local desktop session. Policy is made available to the deviceTRUST Agent using existing Microsoft Active Directory Group Policy Management or file-based. All operations performed by the deviceTRUST Agent are written to the Microsoft Windows Event Log.



Remote Scenario

deviceTRUST consists of two main components when installing in remote environments, the deviceTRUST Agent and the deviceTRUST Client Extension. Both deviceTRUST components can be installed and configured within minutes and can be fully integrated with existing deployment processes and management tools. No additional infrastructure (e.g. a database or a web server) is required for deviceTRUST to be installed in your environment.

Solutions are provided by deviceTRUST® for both traditional and modern Operating Systems (OS). On a traditional OS such as Microsoft Windows, an extensibility framework is available that enables deviceTRUST to send user and device context within the communication channel between the clients and the Remote Desktop Services host. deviceTRUST also provides a solution for more modern OS's, such as Apple iOS, which offer no extensibility framework.

deviceTRUST Agent

This component needs to be installed on the remoting host that delivers the remote session to the users. The following technologies are supported by deviceTRUST: Amazon WorkSpaces, Citrix Virtual Apps and Desktops™ (CVAD), Microsoft Azure Virtual Desktop (AVD), Microsoft Remote Desktop Session Host (RDSH) or VMware Horizon View.

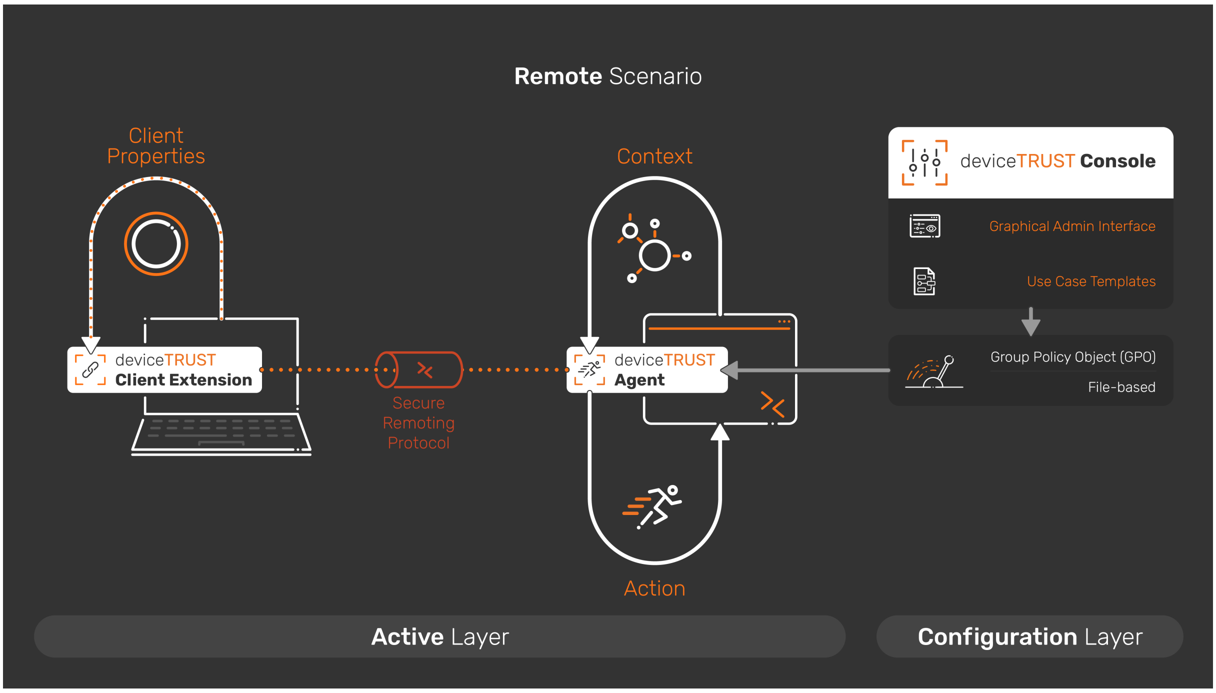
deviceTRUST Client Extension

This component needs to be installed on the remote device which will be used to connect to the remote host delivering the published applications and desktops. It is not required to have deviceTRUST Client Extension installed onto all of your remote devices but recommended to get the full range of context information about the remote device and its user into the users'virtual session.

In the absence of the deviceTRUST Client Extension on the remote device, deviceTRUST delivers the LOCAL_* properties into the users'remote session. The Property Reference describes which properties of the local agent and remote device are available within the users'session.

Architecture - Windows, macOS, Ubuntu, eLux® RP or IGEL OS device

The following diagram details the deviceTRUST architecture when the remote client is installed on a Windows, macOS, Ubuntu, eLux RP or IGEL OS device, with deviceTRUST sending the user and device context information within the communication channel offered by the remoting protocol. Policy is made available to the deviceTRUST Agent using existing Microsoft Active Directory Group Policy Management or file-based. All operations performed by the deviceTRUST Agent are written to the Microsoft Windows Event Log.



Operating System Compatibility

- Apple iOS and iPadOS
- Apple macOS
- IGEL OS 10 and 11
- IGEL OS 12
- Microsoft Windows
- Stratodesk NoTouch
- Ubuntu Desktop
- Unicon eLux

Apple iOS and iPadOS

Compatible operating systems (deviceTRUST® Client Extension):

- Apple iOS 13.x and later
- Apple iPadOS 13.x and later

Compatible technologies:

- Citrix Virtual Apps and Desktops™
- Citrix Cloud™

> **Note:**
>
> iOS 17 support requires deviceTRUST Agent 23.1.200 or later, and deviceTRUST Client Extension for iOS 23.1.200 or later.

Apple macOS

Compatible operating systems (deviceTRUST Client Extension):

- Apple macOS 10.15 and later

Compatible technologies:

- Amazon AppStream
- Amazon WorkSpaces (PCoIP) $_{1}$
- Amazon WorkSpaces (WSP) $_{2}$
- Citrix Virtual Apps™ and Desktops
- Citrix Cloud
- Microsoft Remote Desktop Services (via FreeRDP 2)
- VMware Horizon View (Blast)
- VMware Horizon View (PCoIP)
- VMware Horizon View (RDP)

> **Note:**
>
> {1} Amazon WorkSpaces (PCoIP) no longer allows virtual channels to be loaded on Apple macOS. This can be worked around by installing Amazon WorkSpaces Client v5.3.0.
> {2} Amazon WorkSpaces (WSP) requires that the <code class='language-plaintext highlighter-rouge'>Configure extensions</code> policy is set to <code class='language-plaintext highlighter-rouge'>Server and Client</code> and there are additional software requirements. More information can be found in our <a href='/docs/23-1/kb/general/installation/enabling_dcv_extensions_on_amazon_worlspaces_wsp/'>Enabling DCV extensions on Amazon WorkSpaces WSP</a> knowledge base article.

IGEL OS 10 and 11

Compatible technologies (deviceTRUST Client Extension):

- Amazon WorkSpaces (PCoIP) natively integrated in IGEL OS 11.08.200 and later {1}
- Citrix Virtual Apps and Desktops natively integrated in IGEL OS 10.03.500 and later
- Citrix Cloud natively integrated in IGEL OS 10.03.500 and later

- Microsoft Azure Virtual Desktop (AVD) natively integrated in IGEL OS 11.08.200 and later {1}
- Microsoft Remote Desktop natively integrated in IGEL OS 10.03.500 and later
- VMware Horizon View (Blast) natively integrated in IGEL OS 10.08.230 and later {1}
- VMware Horizon View (PCoIP) natively integrated in IGEL OS 10.08.230 and later {1}
- VMware Horizon View (RDP) natively integrated in IGEL OS 10.08.230 and later {1}

> **Note:**
>
> {1} Compatibility with previous IGEL OS releases is available by contacting deviceTRUST Support.

IGEL OS 12

Compatible technologies (deviceTRUST Client Extension):

- Citrix Virtual Apps and Desktops available for IGEL OS 12.01.120 and later {1}
- Citrix Cloud natively integrated available for IGEL OS 12.01.120 and later {1}
- Microsoft Azure Virtual Desktop (AVD) available for IGEL OS 12.01.120 and later {1}
- VMware Horizon View (Blast) available for IGEL OS 12.01.120 and later {1}
- VMware Horizon View (PCoIP) available for IGEL OS 12.01.120 and later {1}
- VMware Horizon View (RDP) available for IGEL OS 12.01.120 and later {1}

> **Note:**
>
> {1} The deviceTRUST Client Extension can be installed from the IGEL App Portal.

Microsoft Windows

Compatible operating systems (deviceTRUST Agent and Console):

- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later

Compatible operating systems (deviceTRUST Client Extension):

- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later

Compatible technologies:

- Amazon AppStream
- Amazon WorkSpaces (PCoIP)
- Amazon WorkSpaces (WSP) {1}
- Azure Virtual Desktop
- Azure Active Directory

- Citrix Virtual Apps and Desktops
- Citrix Cloud
- Microsoft Remote Desktop Services
- Nice DCV Standalone
- Parallels Remote Application Server
- VMware Horizon View (Blast)
- VMware Horizon View (PCoIP)
- VMware Horizon View (RDP)

> **Note:**
>
> {1} Amazon WorkSpaces (WSP) requires that the `<code class='language-plaintext highlighter-rouge'>Configure extensions</code> policy is set to <code class='language-plaintext highlighter-rouge'>Server and Client</code>` and there are additional software requirements. More information can be found in our Enabling DCV extensions on Amazon WorkSpaces WSP knowledge base article.

Stratodesk NoTouch

Compatible technologies (deviceTRUST Client Extension):

- Citrix Virtual Apps and Desktops on NoTouch OS 3.4.516 and later
- Citrix Cloud on NoTouch OS 3.4.516 and later
- Microsoft Remote Desktop on NoTouch OS 3.4.516 and later
- VMware Horizon View (Blast) on NoTouch OS 3.4.516 and later
- VMware Horizon View (PCoIP) on NoTouch OS 3.4.516 and later
- VMware Horizon View (RDP) on NoTouch OS 3.4.516 and later

Ubuntu Desktop

Compatible operating systems (deviceTRUST Client Extension):

- Ubuntu Desktop 18.04 LTS and later

Compatible technologies:

- Amazon WorkSpaces (PCoIP)
- Citrix Virtual Apps and Desktops
- Citrix Cloud
- Microsoft Remote Desktop Services (via FreeRDP 2)
- VMware Horizon View (Blast)
- VMware Horizon View (PCoIP)
- VMware Horizon View (RDP)

Unicon™ eLux

Compatible technologies (deviceTRUST Client Extension):

- Citrix Virtual Apps and Desktops natively integrated in eLux® RP 6
- Citrix Cloud natively integrated in eLux RP 6
- Microsoft Remote Desktop natively integrated in eLux RP 6

Platform Compatibility

- Amazon Web Services
- Citrix Systems
- Microsoft
- Parallels
- VMware

Amazon Web Services

Compatible technologies:

- Amazon WorkSpaces (PCoIP)
- Amazon WorkSpaces (WSP) {2 & 3}
- Amazon AppStream {2}
- Nice DCV Standalone {2}

Compatible operating systems:

- Apple macOS 10.15 and later
- IGEL OS 11.08.200 and later versions of IGEL OS 11 {1}
- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later
- Ubuntu Desktop 18.04 LTS

> **Note:**
>
> {1} Compatibility with previous IGEL OS releases is available by contacting deviceTRUST® Support. Compatibility with IGEL OS 12 is not yet available.
> {2} Amazon WorkSpaces (WSP), Amazon AppStream and Nice DCV Standalone currently require Microsoft Windows or Apple macOS clients.
> {3} Amazon WorkSpaces (WSP) requires that the `Configure extensions` policy is set to `Server and Client` and there are additional software requirements. More information can

> be found in our Enabling DCV extensions on Amazon WorkSpaces WSP

Citrix Systems

Compatible technologies:

- Citrix Virtual Apps and Desktops™ {1}
- Citrix Cloud™ {1}

Compatible operating systems:

- Apple iOS 13.x and later
- Apple iPadOS 13.x and later
- Apple macOS 10.15 and later
- IGEL OS 10.03.500 and later
- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later
- Stratodesk NoTouch OS 3.4.516 and later
- Ubuntu Desktop 18.04 LTS and later
- Unicon™ eLux RP 6

> **Note:**
>
> {1} Compatibility with HTML5 delivered apps and desktops is not yet available.

Microsoft

Compatible technologies:

- Microsoft Remote Desktop Services
- Azure Virtual Desktop {1}
- Azure Active Directory {2}

Compatible operating systems:

- Apple iOS 13.x and later
- Apple iPadOS 13.x and later
- Apple macOS 10.15 and later (via FreeRDP 2)
- IGEL OS 11.08.200 or later {3}
- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 and later
- Ubuntu Desktop 18.04 LTS and later (via FreeRDP 2)

> **Note:**
>
> {1} Compatibility with Azure Virtual Desktop is available on Microsoft Windows and IGEL OS only.
> {2} Compatibility with Azure Active Directory provided by deviceTRUST Agent on Microsoft Windows only.
> {3} Compatibility with previous IGEL OS releases is available by contacting deviceTRUST Support. Compatibility with IGEL OS 12 is currently limited to Azure Virtual Desktop.

Parallels

Compatible technologies:

- Parallels Remote Application Server

Compatible operating systems:

- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later

VMware

Compatible technologis:

- VMware Horizon View (Blast)
- VMware Horizon View (PCoIP)
- VMware Horizon View (RDP)

Compatible operating systems:

- Apple iOS 13.x and later
- Apple iPadOS 13.x and later
- Apple macOS 10.15 and later
- IGEL OS 10.08.230 or later {1}
- Microsoft Windows 10 and later
- Microsoft Windows Server 2012 R2 and later
- Stratodesk NoTouch OS 3.4.516 and later
- Ubuntu Desktop 18.04 LTS and later

> **Note:**
>
> {1} Compatibility with previous IGEL OS releases is available by contacting deviceTRUST Support.

Getting Started

deviceTRUST® requires some simple but essential configuration steps to be performed to enable deviceTRUST functionality for your remoting environments or for your local devices.

Scenario: Remote

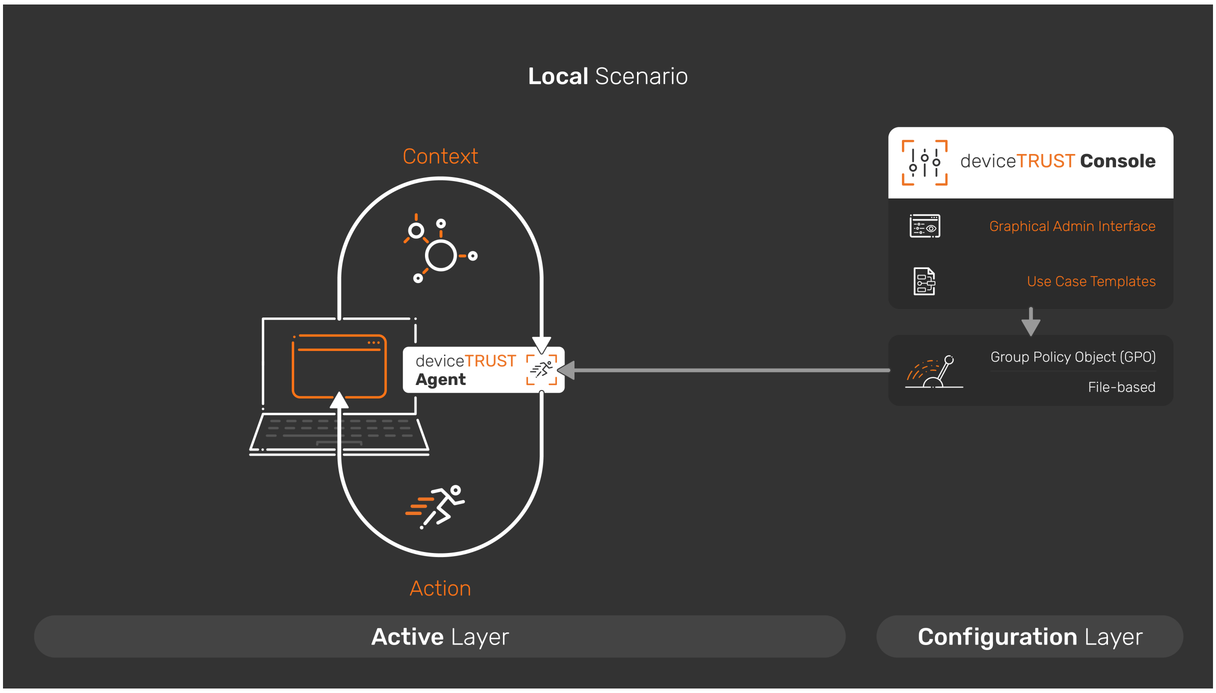
In remote scenarios, deviceTRUST transports the context information from the users remote device into the virtual session where the configuration is enforced. Please visit Getting Started for Remote Devices to begin the guide.

Scenario: Local

In local scenarios, deviceTRUST collects context information and executes actions locally. Please visit Getting Started for Local Devices to begin the guide.

Getting Started for Local

deviceTRUST® requires some simple but essential configuration steps to be performed to enable deviceTRUST functionality for your local devices. We will guide you step-by-step through simple deviceTRUST installation and configuration steps to enable deviceTRUST with an unauthorized USB drives use case for your local devices.



We will perform the following steps:

- Step 1: Download the deviceTRUST setup binaries
- Step 2: Install the deviceTRUST Agent
- Step 3: Install the deviceTRUST Console
- Step 4: Enter your deviceTRUST License
- Step 5: Enable the Unauthorized USB Drive use case
- Step 6: Test the Unauthorized USB Device use case

Step 1: Download the deviceTRUST setup binaries

The latest deviceTRUST software can be found on our Download page and your personalized license can be found within your product license certificate.

Step 2: Install the deviceTRUST Agent

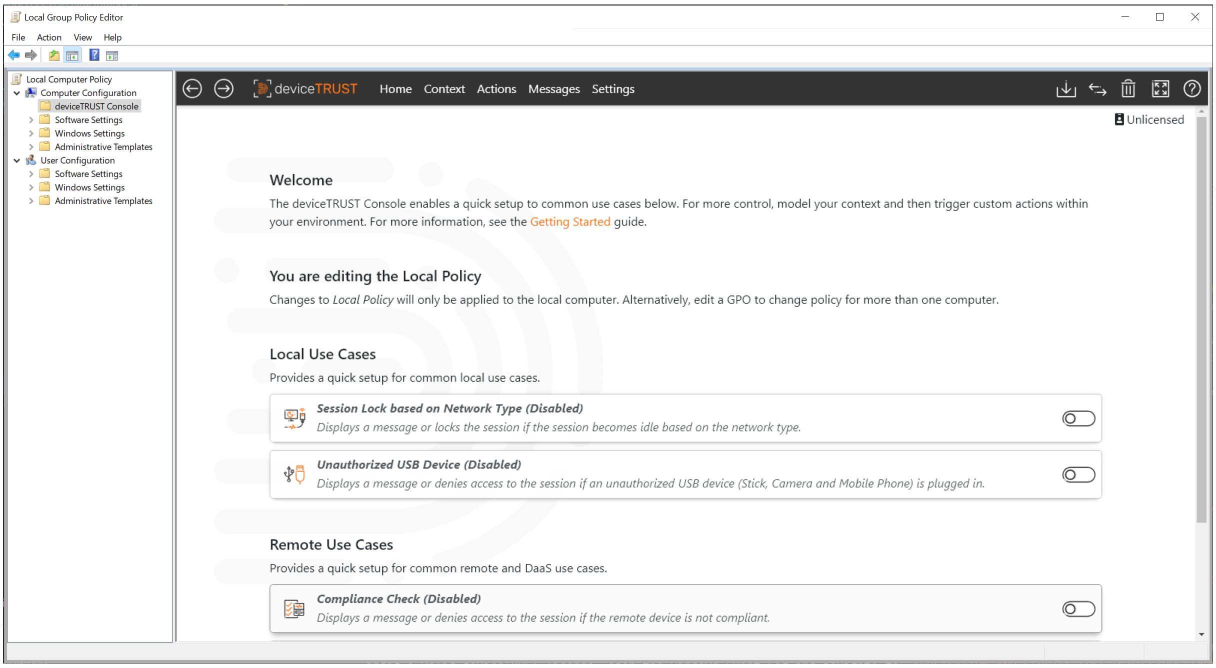
Start the installation of the deviceTRUST Agent on your local device. Follow the steps in the section Installing the Agent to complete the installation.

Step 3: Install the deviceTRUST Console

To configure and to apply contextual security policies to the deviceTRUST Agent you need to use the deviceTRUST Console. The deviceTRUST Console supports various ways to provide the contextual security policies to the deviceTRUST Agent. Those options are using the Local Policy Editor, a Group Policy Object (GPO) or file-based.

Within the Getting Started Guide, for simplicity, we use the Local Policy Editor to quickly and efficiently create, edit, and use contextual security policies. Follow the steps in the section Installing the Console to complete the installation.

The deviceTRUST Console includes a node within the Local Policy Editor `COMPUTER CONFIGURATION\DEVICETRUST CONSOLE` which can be used to model the context of a user, and then act on changes to that context by triggering custom actions within your environment.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

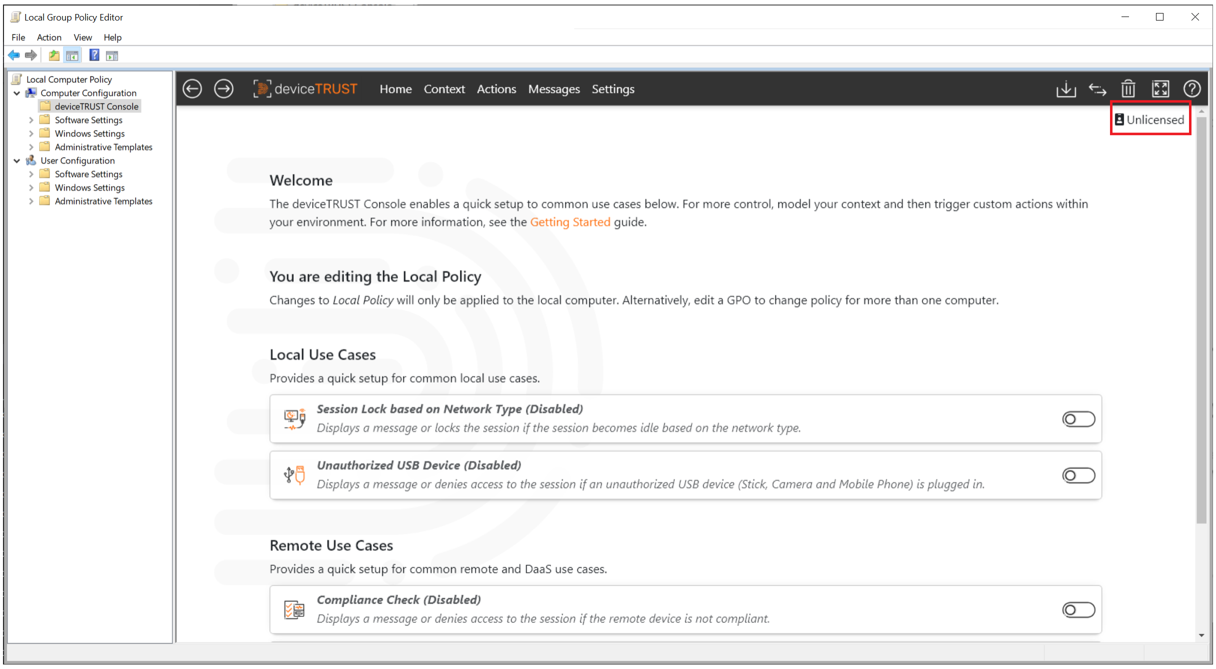
Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Step 4: Enter your deviceTRUST License

To add the license into the deviceTRUST contextual security policy open the Local Policy Editor and navigate to `DEVICETRUST CONSOLE` and click on the `UNLICENSED` link on the homepage.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

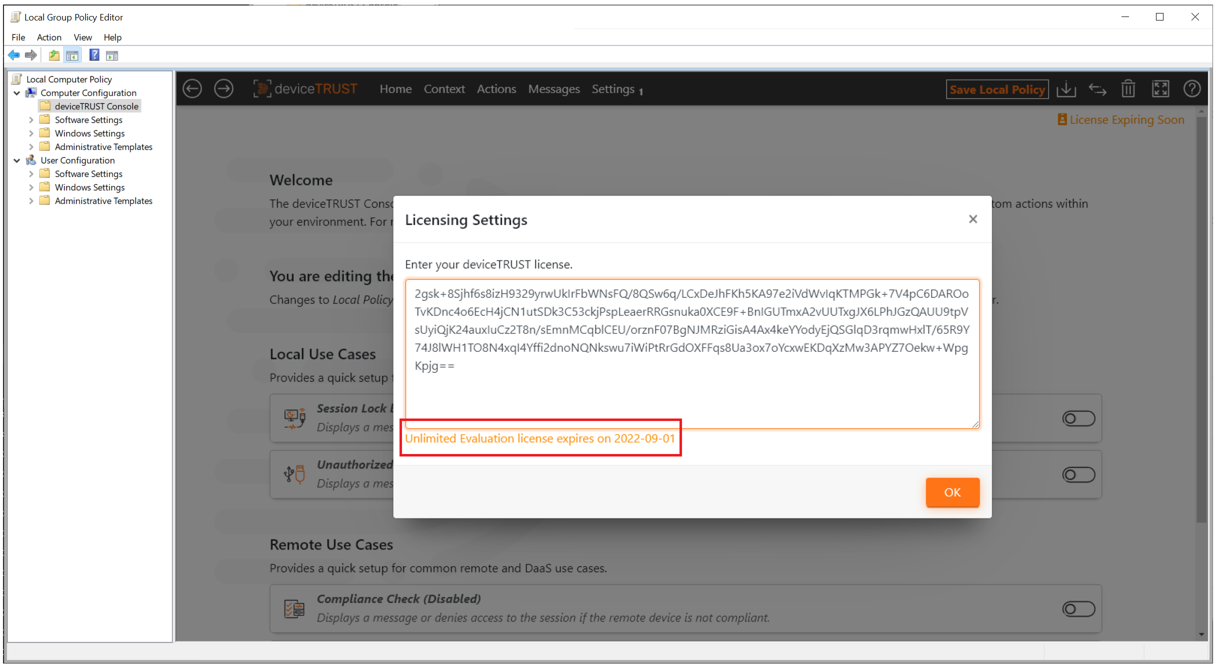
Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Enter your deviceTRUST license and make sure it is valid. Close the license editor with `OK` and click on `SAVE TO LOCAL COMPUTER POLICY` in the top right toolbar.



Licensing Settings

Enter your deviceTRUST license.

```
2gsk+8Sjhf6s8izH9329yrwUkIrFbWNsFQ/8QSw6q/LCxDeJhFKh5KA97e2iVdWvIqKTMPGk+7V4pC6DAROo
TvKDnc4o6EcH4jCN1utSDk3C53ckjPspLeaerRRGsnuka0XCE9F+BnIGUTmxA2vUUTxgJX6LPhJGzQAUU9tpV
sUyiQjK24auxIuCz2T8n/sEmnMCqblCEU/orznF07BgNJMRziGisA4Ax4keYYodyEjQSGlqD3rqmwHxlT/65R9Y
74J8lWH1TO8N4xqI4Yffi2dnoNQNkswu7iWiPtRrGdOXFFqs8Ua3ox7oYcxwEKDqXzMw3APYZ7Oekw+Wpg
Kpjg==
```

Unlimited Evaluation license expires on 2022-09-01

OK

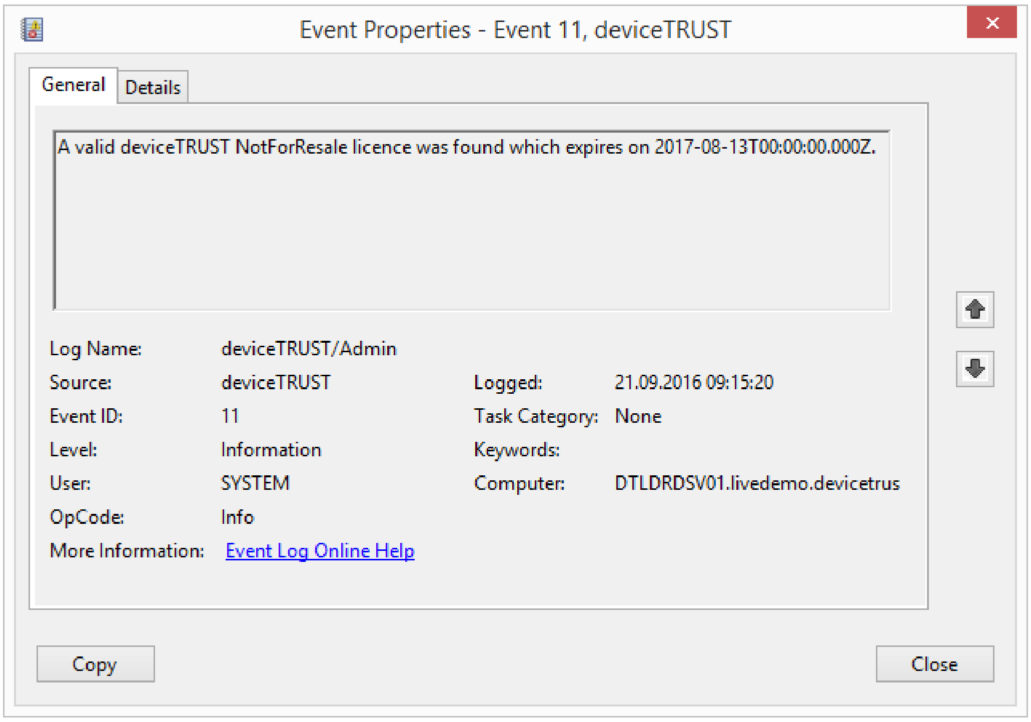
Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)

Displays a message or denies access to the session if the remote device is not compliant.

deviceTRUST is now enabled and will work for all users except local administrators connecting to that remoting or DaaS host system with deviceTRUST Agent installed. To check if you have added a valid deviceTRUST license, open the Windows Event Log and navigate to `APPLICATION AND SERVICE LOGS\DEVICETRUST\ADMIN` and check for the existence of event ID 11 which states that your deviceTRUST license is valid.

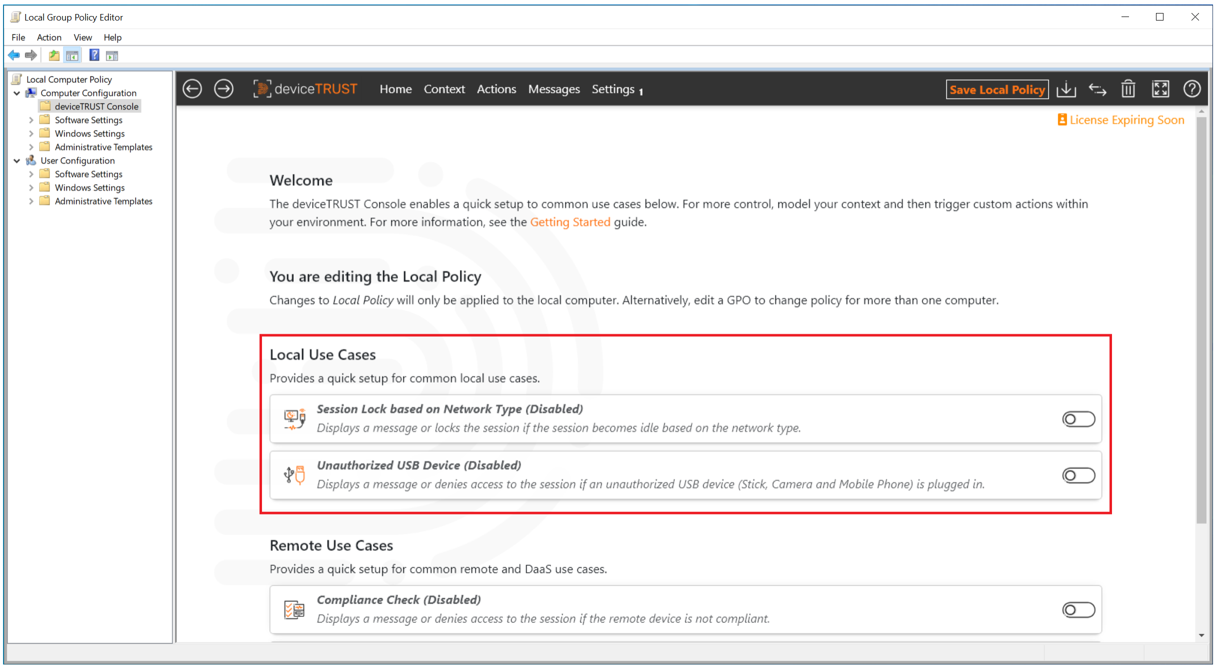


A valid deviceTRUST NotForResale licence was found which expires on 2017-08-13T00:00:00.000Z.

Log Name:	deviceTRUST/Admin		
Source:	deviceTRUST	Logged:	21.09.2016 09:15:20
Event ID:	11	Task Category:	None
Level:	Information	Keywords:	
User:	SYSTEM	Computer:	DTLDRDSV01.livedemo.devicetrus
OpCode:	Info		
More Information:	Event Log Online Help		

Step 5: Enable the Unauthorized USB Drive use case

We will use the deviceTRUST Console to create a contextual security policy that makes access to the session dependent on whether the USB device being used has been authorized. The deviceTRUST Console includes a set of use cases which can be used to quickly implement a use case. Launch the deviceTRUST Console and navigate to `LOCAL USE CASES` on the homepage.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

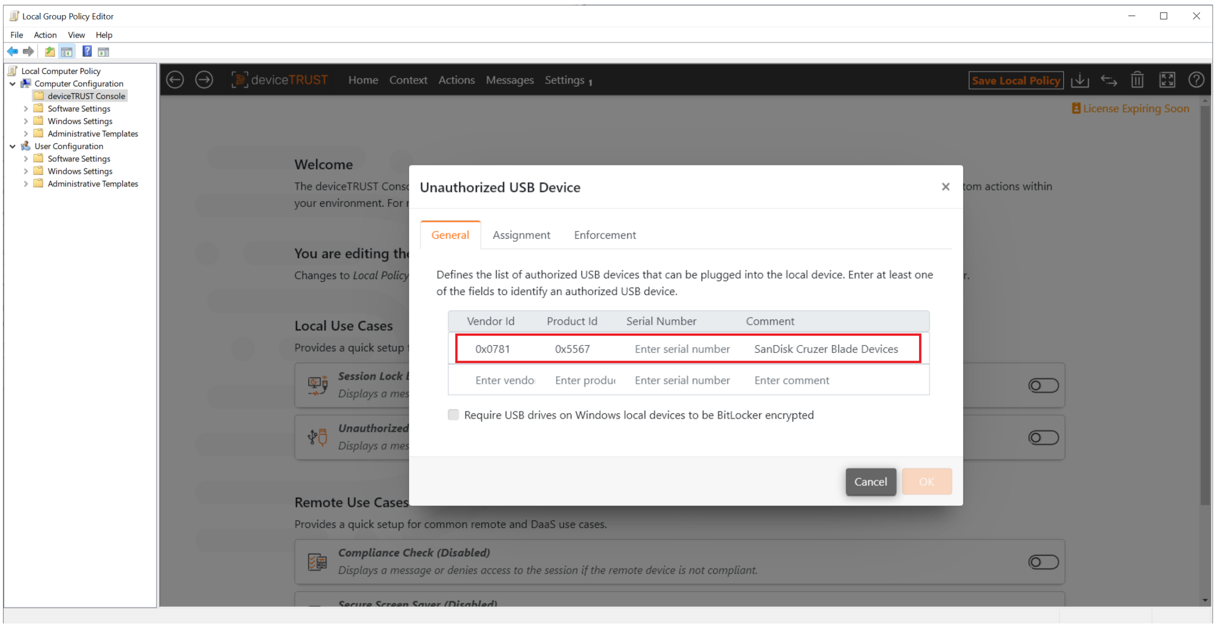
Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Select the `UNAUTHORIZED USB DEVICE` use case and add authorized USB devices on the `GENERAL` tab into the list of authorized USB devices.



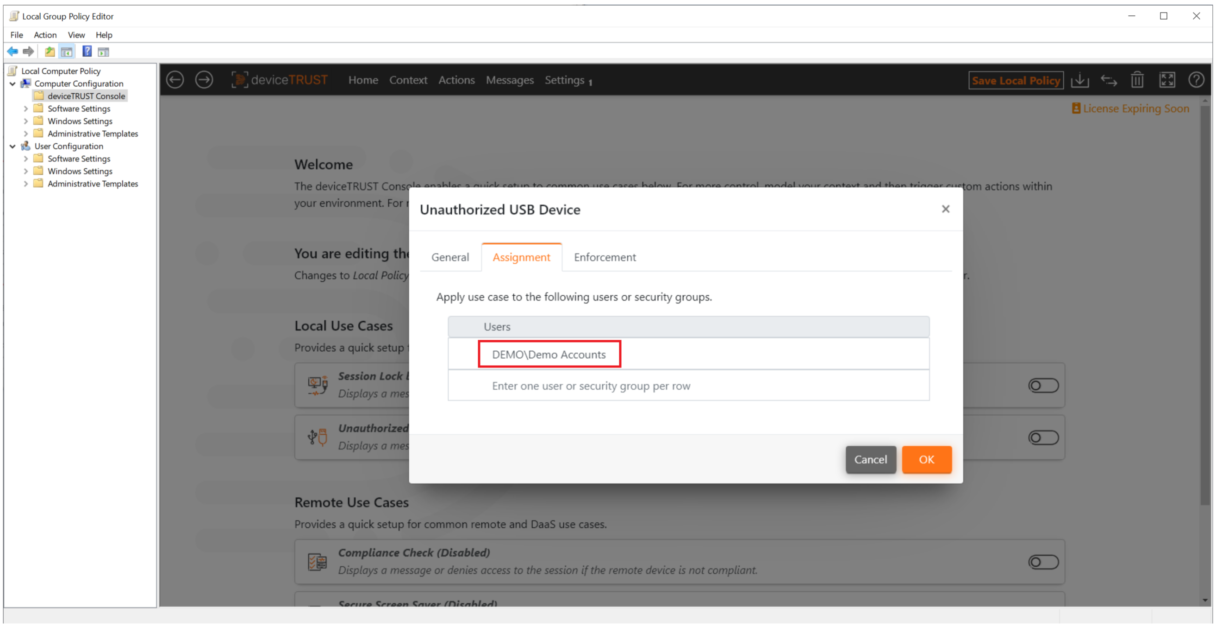
Unauthorized USB Device

Defines the list of authorized USB devices that can be plugged into the local device. Enter at least one of the fields to identify an authorized USB device.

Vendor Id	Product Id	Serial Number	Comment
0x0781	0x5567	Enter serial number	SanDisk Cruzer Blade Devices
Enter vendo	Enter produ	Enter serial number	Enter comment

□ Require USB drives on Windows local devices to be BitLocker encrypted

Click on the `ASSIGNMENT` configuration tab and add `USERS` and / or `SECURITY GROUPS` to apply the use case for.



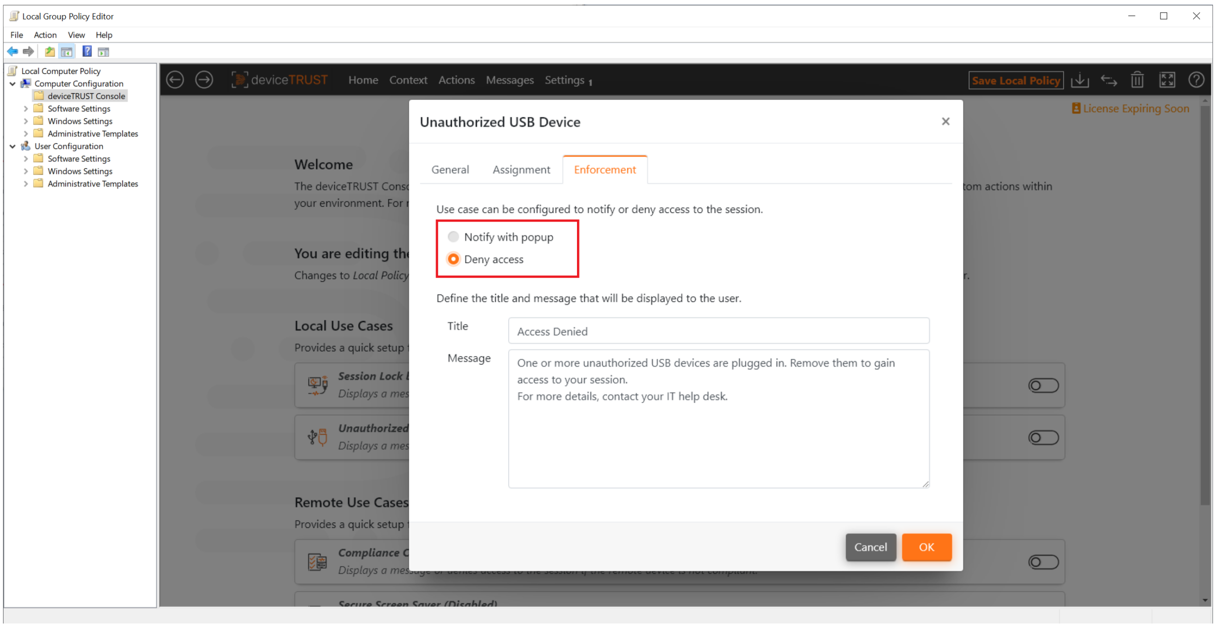
Unauthorized USB Device

General | Assignment | Enforcement

Apply use case to the following users or security groups.

Users
DEMO\Demo Accounts
Enter one user or security group per row

Click on the ENFORCEMENT configuration tab and select DENY ACCESS.



Unauthorized USB Device

General | Assignment | Enforcement

Use case can be configured to notify or deny access to the session.

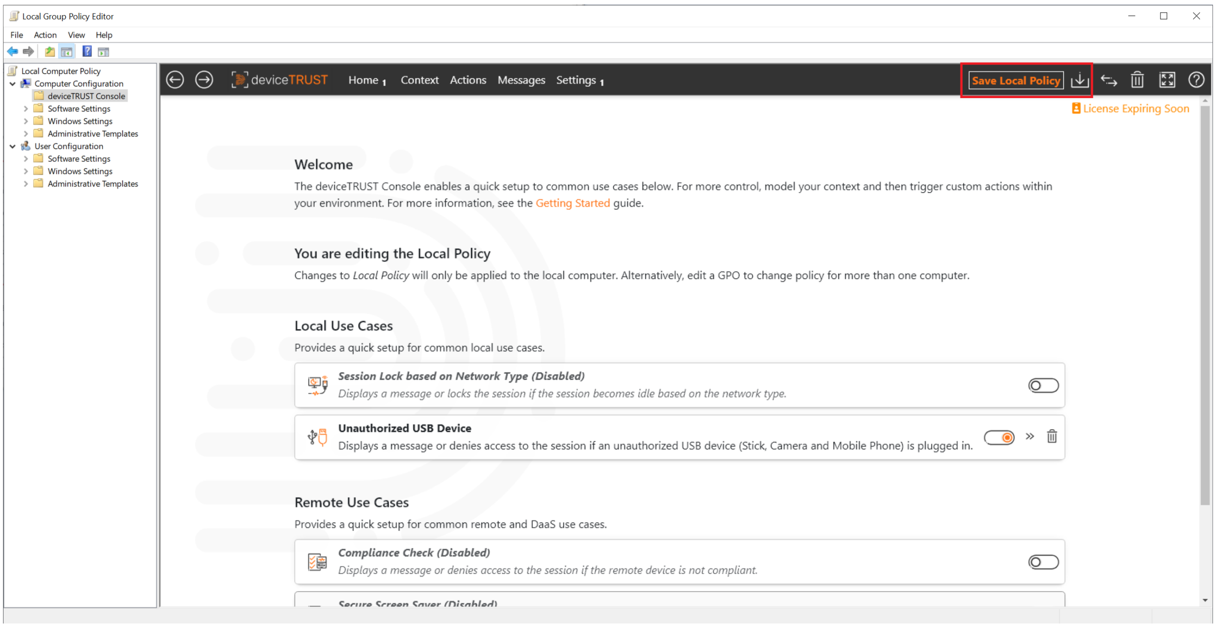
○ Notify with popup
● Deny access

Define the title and message that will be displayed to the user.

Title: Access Denied

Message: One or more unauthorized USB devices are plugged in. Remove them to gain access to your session.
For more details, contact your IT help desk.

Click on SAVE TO LOCAL COMPUTER POLICY in the top right toolbar to save the unauthorized USB device use case to the local computer policy.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

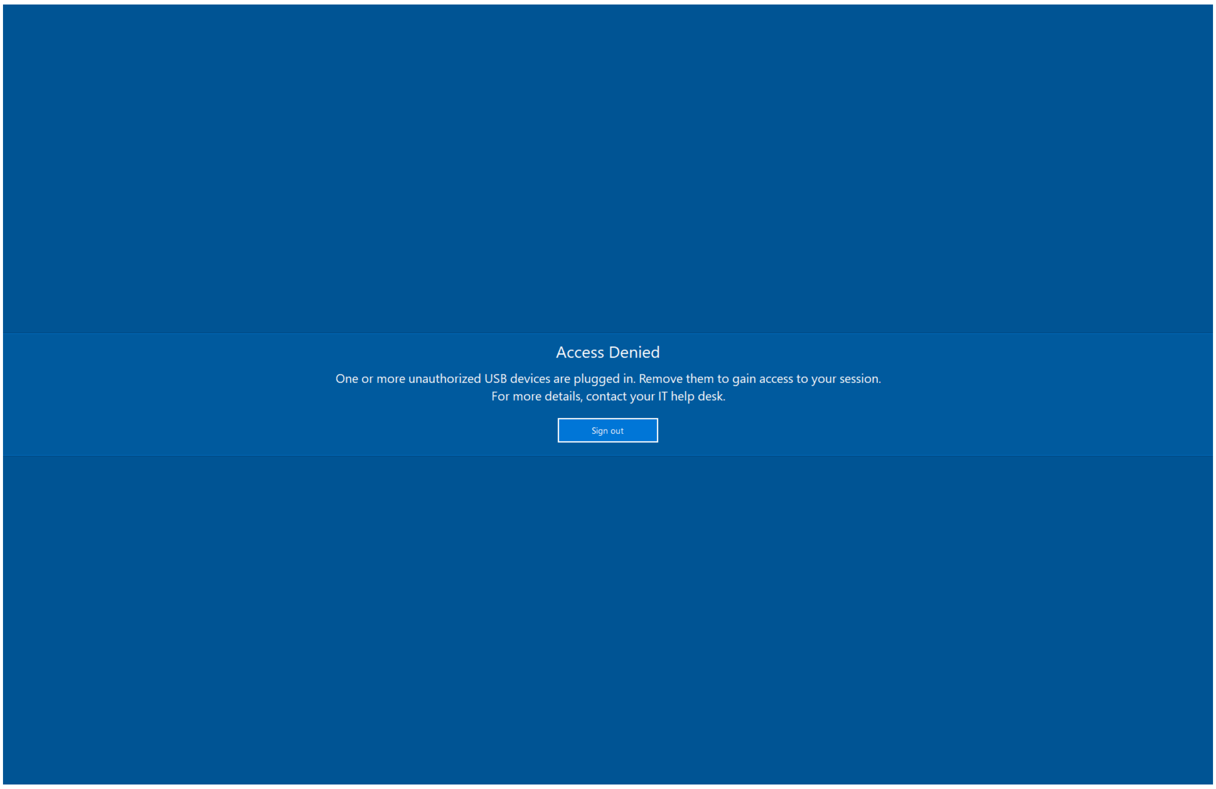
Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Secure Screen Saver (Disabled)

Step 6: Test the Unauthorized USB Device use case

Sign in with a non-administrative user account to the local device and then plug in an authorized USB device at runtime. The authorized USB device is displayed in Windows Explorer and can be used. Now plug in an unauthorized USB device in addition or exclusively to see how deviceTRUST can easily and dynamically control access to the session depending on the USB device in use.

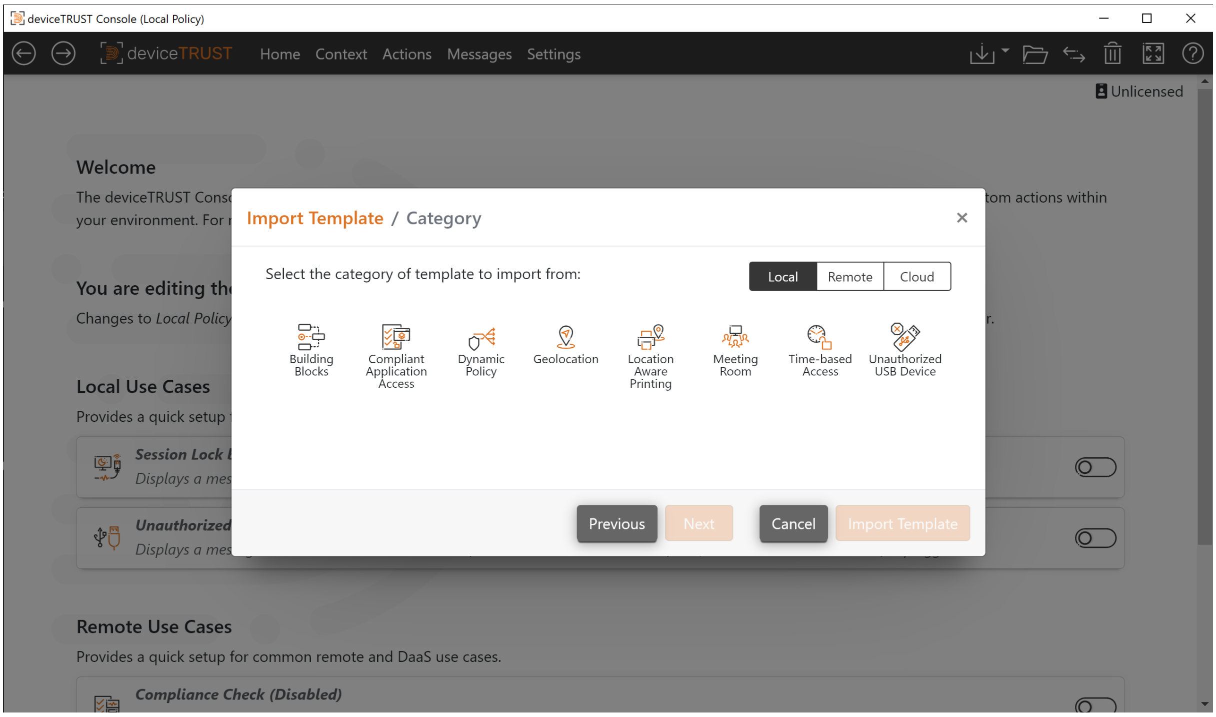


Access Denied

One or more unauthorized USB devices are plugged in. Remove them to gain access to your session.
For more details, contact your IT help desk.

Next steps

You have now successfully implemented your first use case with deviceTRUST for your local devices. Feel free to check out our additional use cases provided on the deviceTRUST Console homepage under `LOCAL USE CASES`. In addition, the deviceTRUST Console gives you access to many more configuration Templates for a wide variety of use cases.



Welcome

The deviceTRUST Cons… …tom actions within your environment. For r…

You are editing the…

Changes to *Local Policy* … …r.

Local Use Cases

Provides a quick setup f…

Session Lock L…
Displays a mes…

Unauthorized…
Displays a mes…

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)

Import Template / Category

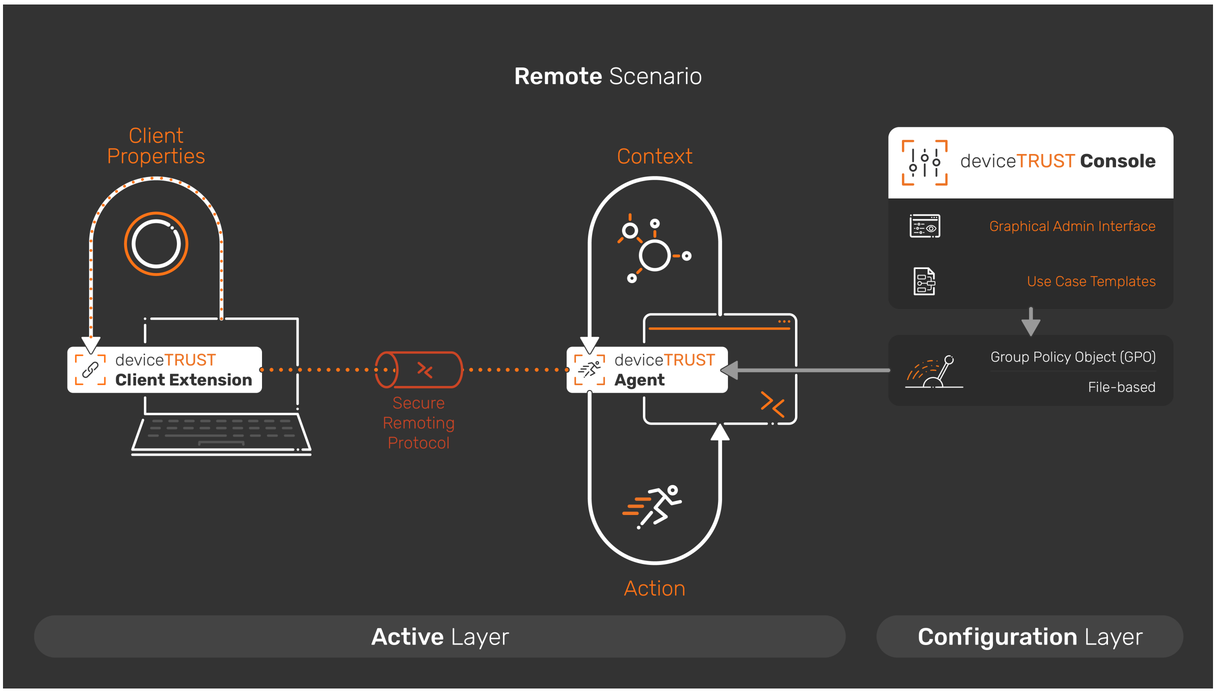
Select the category of template to import from:

Troubleshooting

If your deviceTRUST installation or configuration does not work as expected, you can use the Troubleshooting guide to start troubleshooting.

Getting Started for Remote

deviceTRUST® requires some simple but essential configuration steps to be performed to enable deviceTRUST functionality for your remote environments. We will guide you step-by-step through simple deviceTRUST installation and configuration steps to enable deviceTRUST with a compliance check use case within your remote environment.



We will perform the following steps:

- Step 1: Download the deviceTRUST setup binaries
- Step 2: Install the deviceTRUST Agent
- Step 3: Install the deviceTRUST Console
- Step 4: Enter your deviceTRUST License
- Step 5: Install the deviceTRUST Client Extension on a Microsoft Windows device
- Step 6: Enable the Compliance Check use case
- Step 7: Check that access is denied when the deviceTRUST Client Extension is not installed
- Step 8: Test the Compliance Check use case from a Microsoft Windows device

Step 1: Download the deviceTRUST setup binaries

The latest deviceTRUST software can be found on our Download page and your personalized license can be found within your product license certificate.

Step 2: Install the deviceTRUST Agent

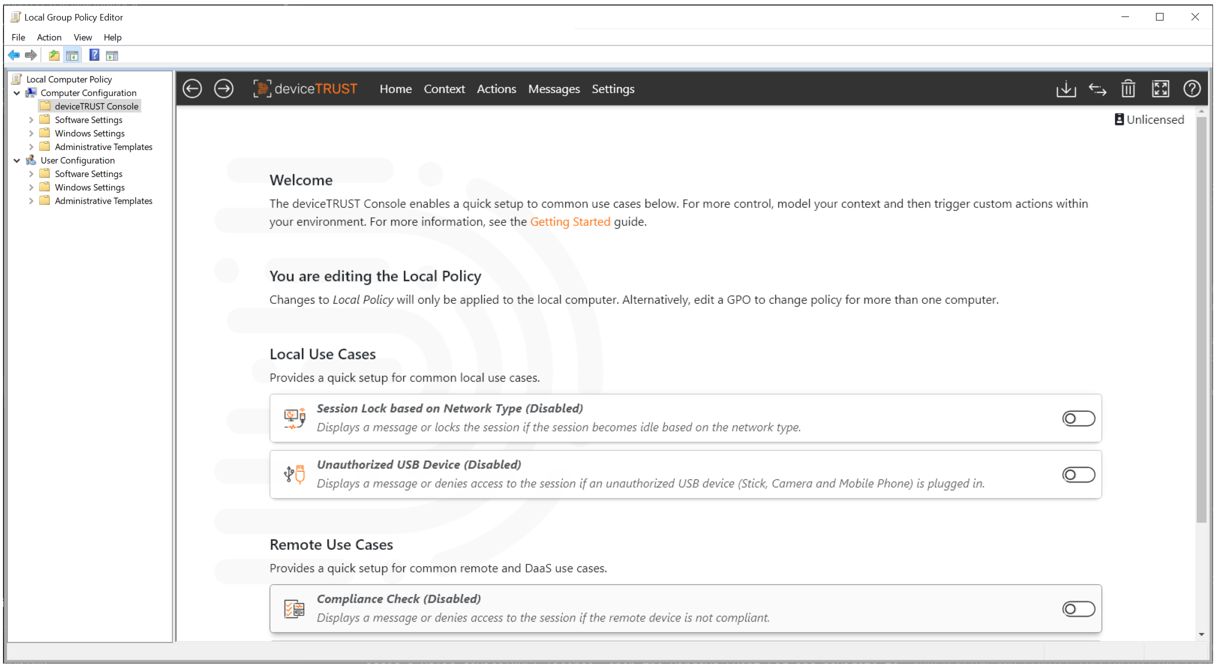
Start the installation of the deviceTRUST Agent on your remoting or DaaS host system, which can be Amazon WorkSpaces, Citrix Virtual Apps and Desktops (CVAD), Microsoft Azure Virtual Desktop (AVD), Microsoft Remote Desktop Session Host (RDSH) or VMware Horizon View. Follow the steps in the section Installing the Agent to complete the installation.

Step 3: Install the deviceTRUST Console

To configure and to apply contextual security policies to the deviceTRUST Agent you need to use the deviceTRUST Console. The deviceTRUST Console supports various ways to provide the contextual security policies to the deviceTRUST Agent. Those options are using the Local Policy Editor, a Group Policy Object (GPO) or file-based.

Within the Getting Started Guide, for simplicity, we use the Local Policy Editor to quickly and efficiently create, edit, and use contextual security policies. Follow the steps in the section Installing the Console to complete the installation.

The deviceTRUST Console includes a node within the Local Policy Editor `COMPUTER CONFIGURATION\DEVICETRUST CONSOLE` which can be used to model the context of a user, and then act on changes to that context by triggering custom actions within your environment.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

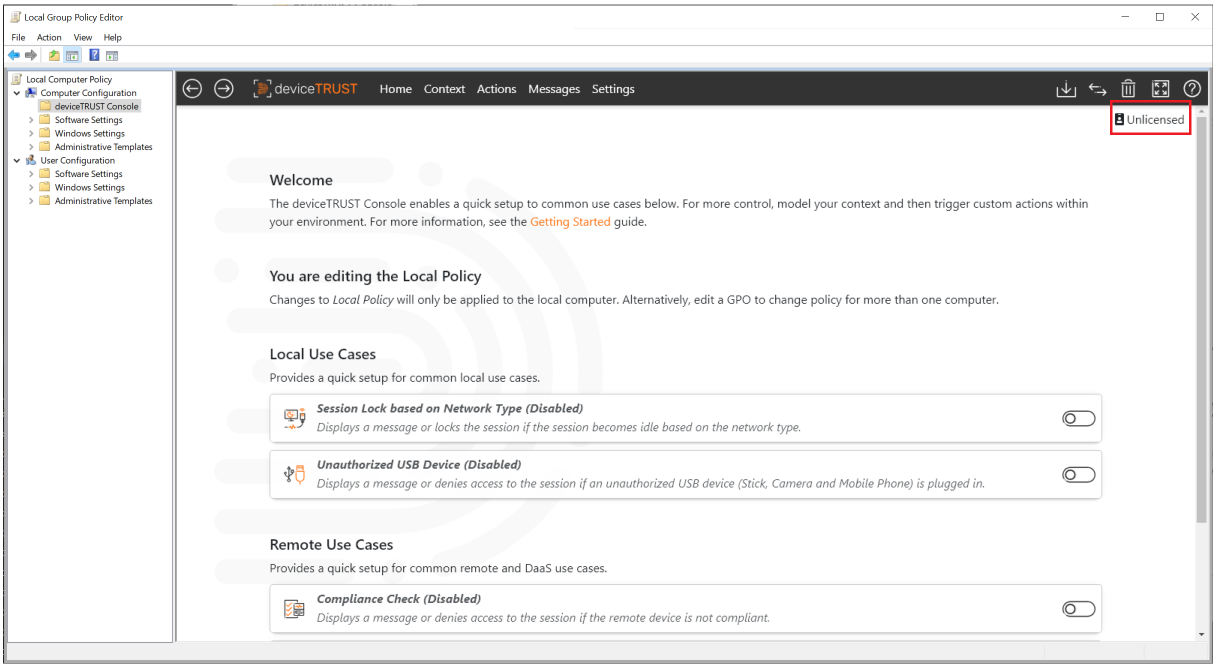
Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Step 4: Enter your deviceTRUST License

To add the license into the deviceTRUST contextual security policy open the Local Policy Editor and navigate to `DEVICETRUST CONSOLE` and click on the `UNLICENSED` link on the homepage.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

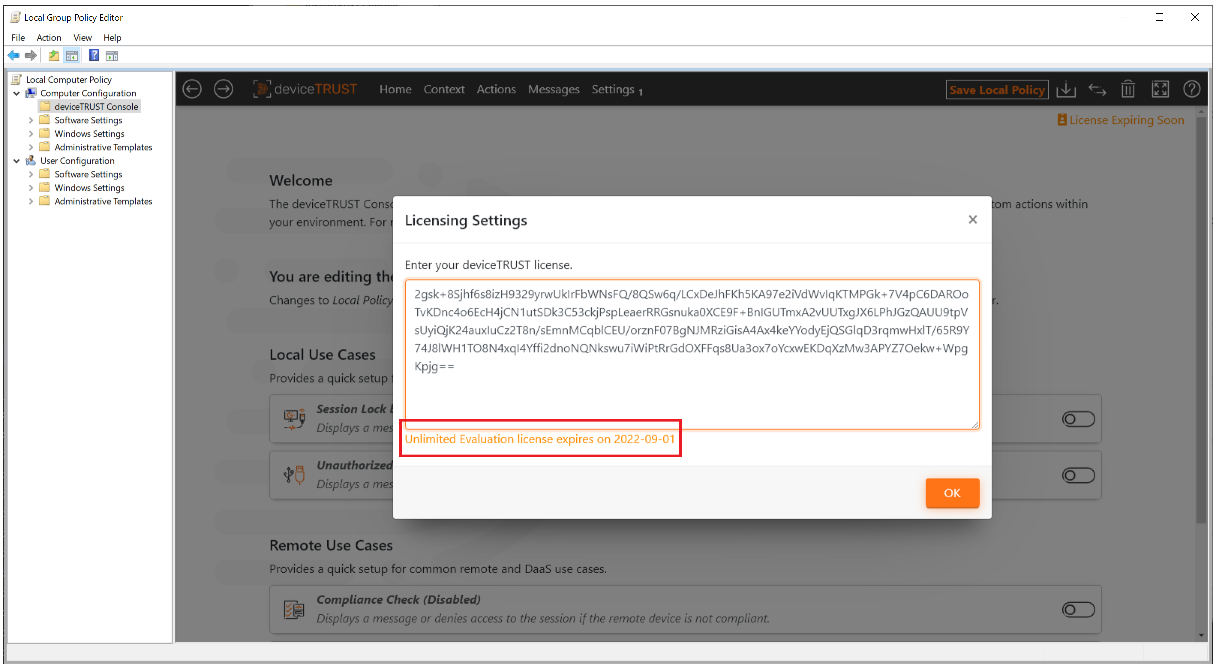
Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

Enter your deviceTRUST license and make sure it is valid. Close the license editor with `OK` and click on `SAVE TO LOCAL COMPUTER POLICY` in the top right toolbar.



Licensing Settings

Enter your deviceTRUST license.

```
2gsk+8Sjhf6s8izH9329yrwUkIrFbWNsFQ/8QSw6q/LCxDeJhFKh5KA97e2iVdWvIqKTMPGk+7V4pC6DAROo
TvKDnc4o6EcH4jCN1utSDk3C53ckjPspLeaerRRGsnuka0XCE9F+BnIGUTmxA2vUUTxgJX6LPhJGzQAUU9tpV
sUyiQjK24auxIuCz2T8n/sEmnMCqblCEU/orznF07BgNJMRziGisA4Ax4keYYodyEjQSGlqD3rqmwHxlT/65R9Y
74J8lWH1TO8N4xqI4Yffi2dnoNQNkswu7iWiPtRrGdOXFFqs8Ua3ox7oYcxwEKDqXzMw3APYZ7Oekw+Wpg
Kpjg==
```

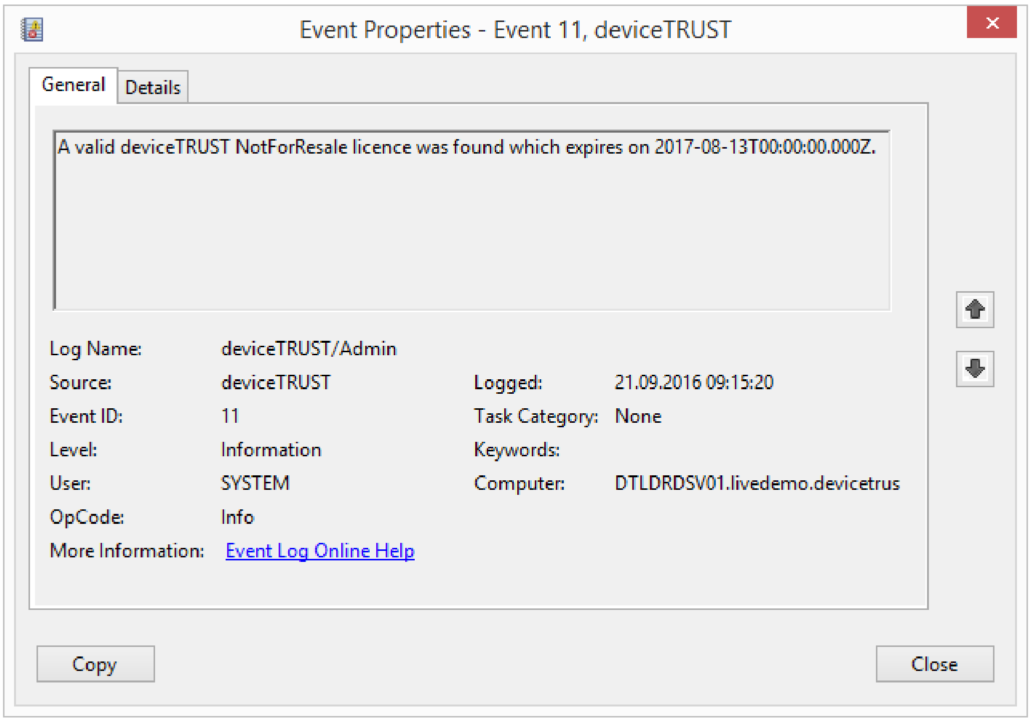
Unlimited Evaluation license expires on 2022-09-01

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

deviceTRUST is now enabled and will work for all users except local administrators connecting to that remoting or DaaS host system with deviceTRUST Agent installed. To check if you have added a valid deviceTRUST license, open the Windows Event Log and navigate to `APPLICATION AND SERVICE LOGS\DEVICETRUST\ADMIN` and check for the existence of event ID 11 which states that your deviceTRUST license is valid.



A valid deviceTRUST NotForResale licence was found which expires on 2017-08-13T00:00:00.000Z.

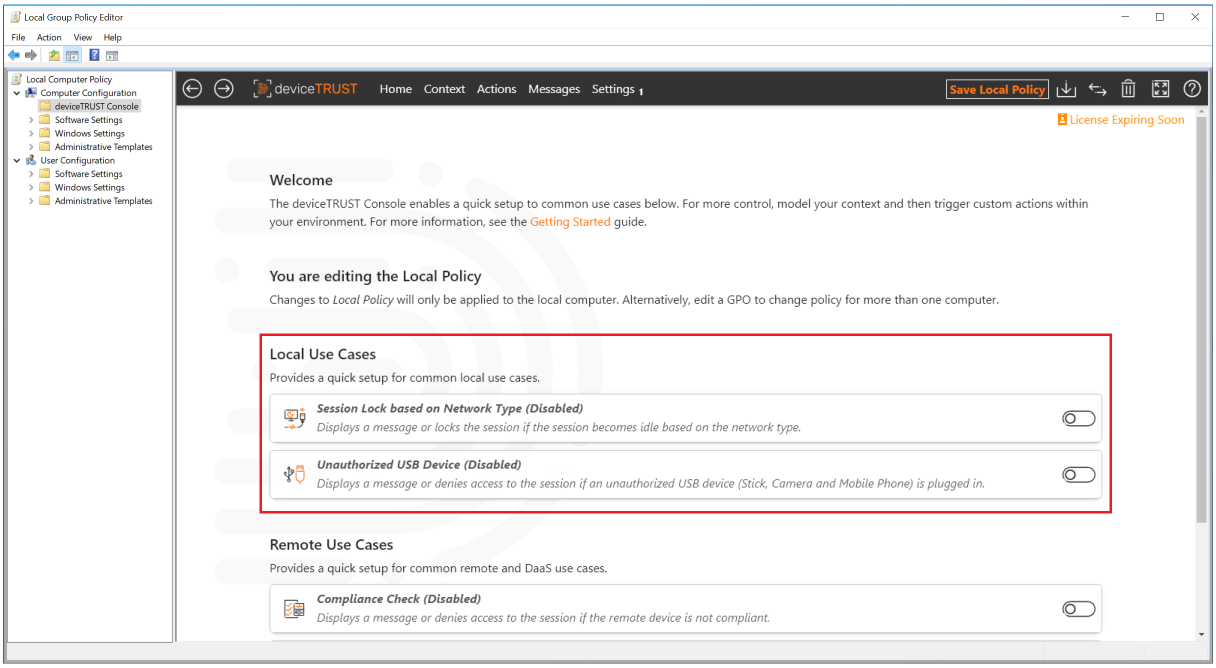
Log Name: deviceTRUST/Admin
Source: deviceTRUST
Logged: 21.09.2016 09:15:20
Event ID: 11
Task Category: None
Level: Information
Keywords:
User: SYSTEM
Computer: DTLDRDSV01.livedemo.devicetrus
OpCode: Info
More Information: Event Log Online Help

Step 5: Install the deviceTRUST Client Extension on a Microsoft Windows device

Within the Getting Started Guide, for simplicity, we will only install the deviceTRUST Client Extension on a Microsoft Windows device. Other device operating systems are also supported and an overview of how to install the deviceTRUST Client Extension on the particular operating system can be found on the Installation Client Extension page. Now follow the steps in the section Installing the Client Extension on Microsoft Windows device to complete the installation.

Step 6: Enable the Compliance Check use case

We will use the deviceTRUST Console to create a contextual security policy which controls access to the session depending upon the compliance state of the remote device. The deviceTRUST Console includes a set of use cases which can be used to quickly implement a use case. Launch the deviceTRUST Console and navigate to `REMOTE USE CASES` on the homepage.



Welcome

The deviceTRUST Console enables a quick setup to common use cases below. For more control, model your context and then trigger custom actions within your environment. For more information, see the Getting Started guide.

You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)

Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device (Disabled)

Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

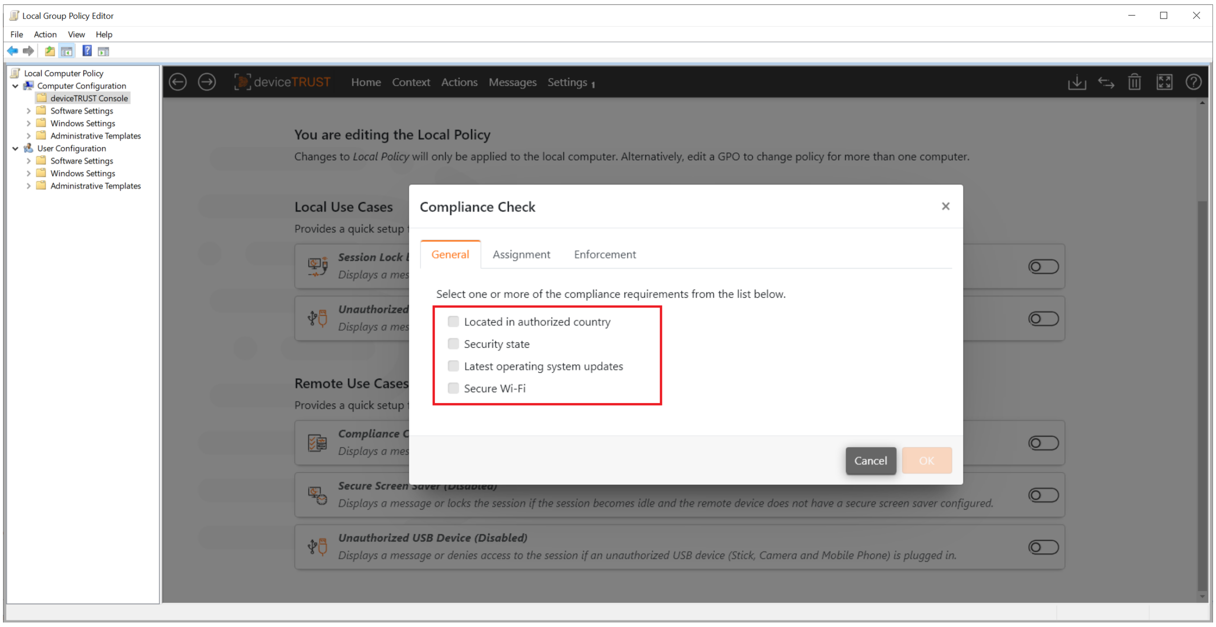
Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)

Displays a message or denies access to the session if the remote device is not compliant.

Select the `COMPLIANCE CHECK` use case, select on the `GENERAL` tab all options to be included in the compliance check.



You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Compliance Check

Select one or more of the compliance requirements from the list below.

□ Located in authorized country
□ Security state
□ Latest operating system updates
□ Secure Wi-Fi

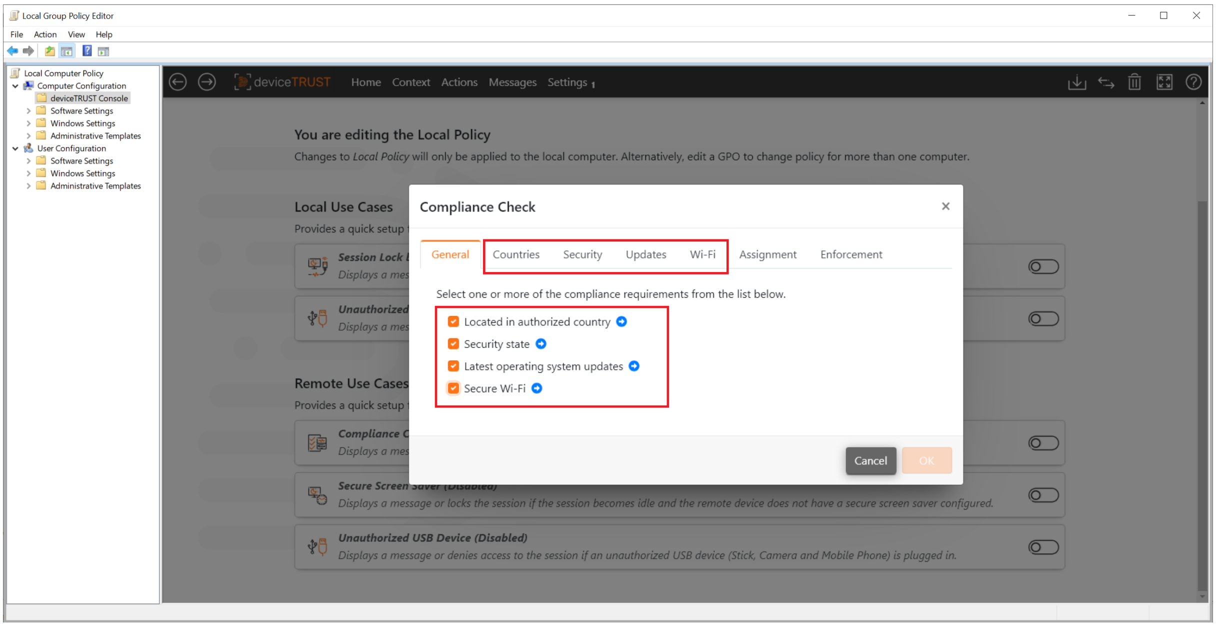
Secure Screen Saver (Disabled)

Displays a message or locks the session if the session becomes idle and the remote device does not have a secure screen saver configured.

Unauthorized USB Device (Disabled)

Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

New configuration tabs will become visible.



You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

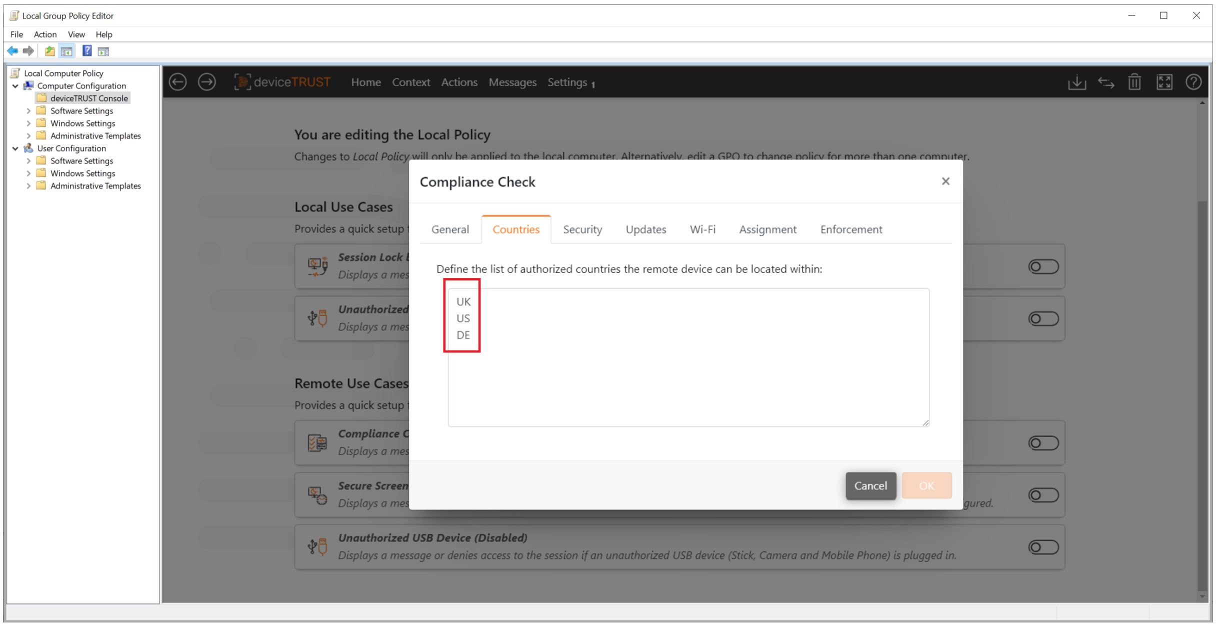
Compliance Check

General | Countries | Security | Updates | Wi-Fi | Assignment | Enforcement

Select one or more of the compliance requirements from the list below.

- ☑ Located in authorized country
- ☑ Security state
- ☑ Latest operating system updates
- ☑ Secure Wi-Fi

Click on the COUNTRY configuration tab and add all authorized countries using ISO 3166-1 Alpha-2 code.



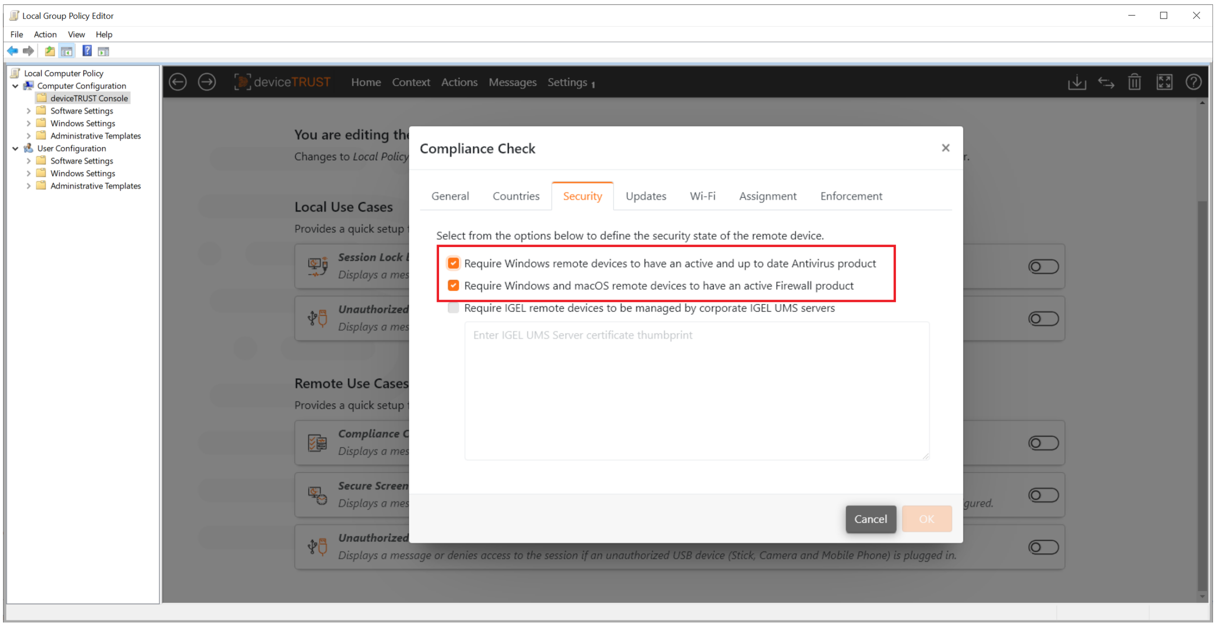
Compliance Check

General | Countries | Security | Updates | Wi-Fi | Assignment | Enforcement

Define the list of authorized countries the remote device can be located within:

```
UK
US
DE
```

Click on the SECURITY configuration tab and enable the REQUIRE WINDOWS REMOTE DEVICES TO HAVE AN ACTIVE AND UP TO DATE ANTIVIRUS PRODUCT and REQUIRE WINDOWS AND MACOS REMOTE DEVICES TO HAVE AN ACTIVE FIREWALL PRODUCT options.



Compliance Check

Select from the options below to define the security state of the remote device.

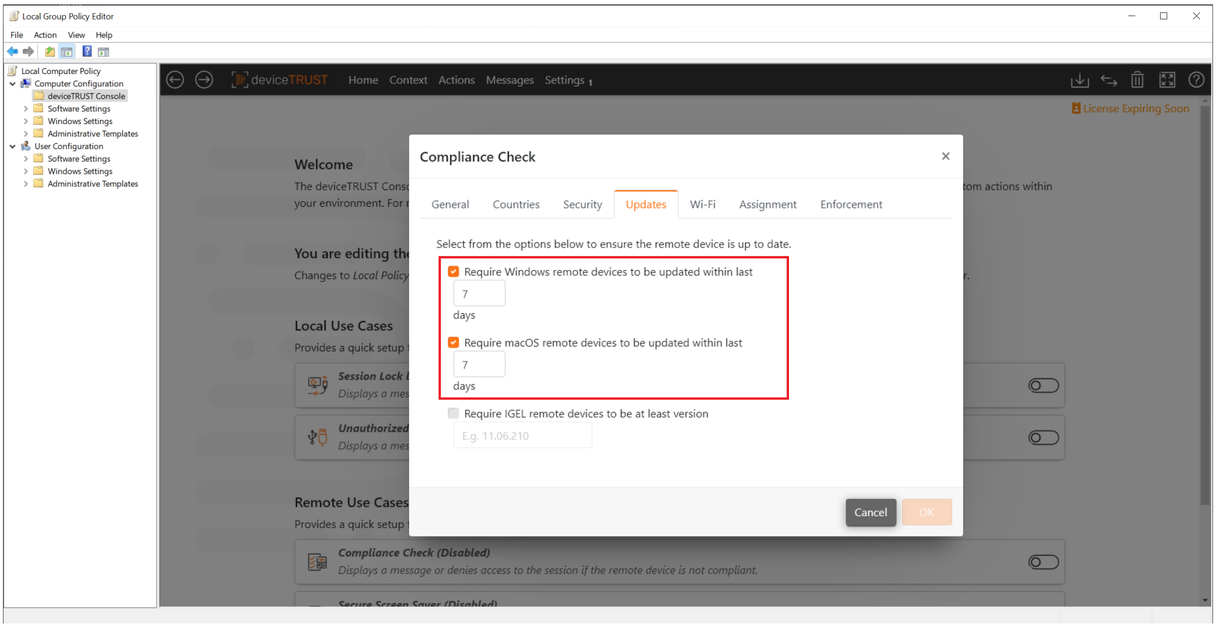
☑ Require Windows remote devices to have an active and up to date Antivirus product

☑ Require Windows and macOS remote devices to have an active Firewall product

□ Require IGEL remote devices to be managed by corporate IGEL UMS servers

Enter IGEL UMS Server certificate thumbprint

Click on the `UPDATES` configuration tab and enable the `REQUIRE WINDOWS REMOTE DEVICES TO BE UPDATED WITHIN THE LAST 7 DAYS` and `REQUIRE MACOS REMOTE DEVICES TO BE UPDATED WITHIN THE LAST 7 DAYS` options.



Compliance Check

Select from the options below to ensure the remote device is up to date.

☑ Require Windows remote devices to be updated within last

7

days

☑ Require macOS remote devices to be updated within last

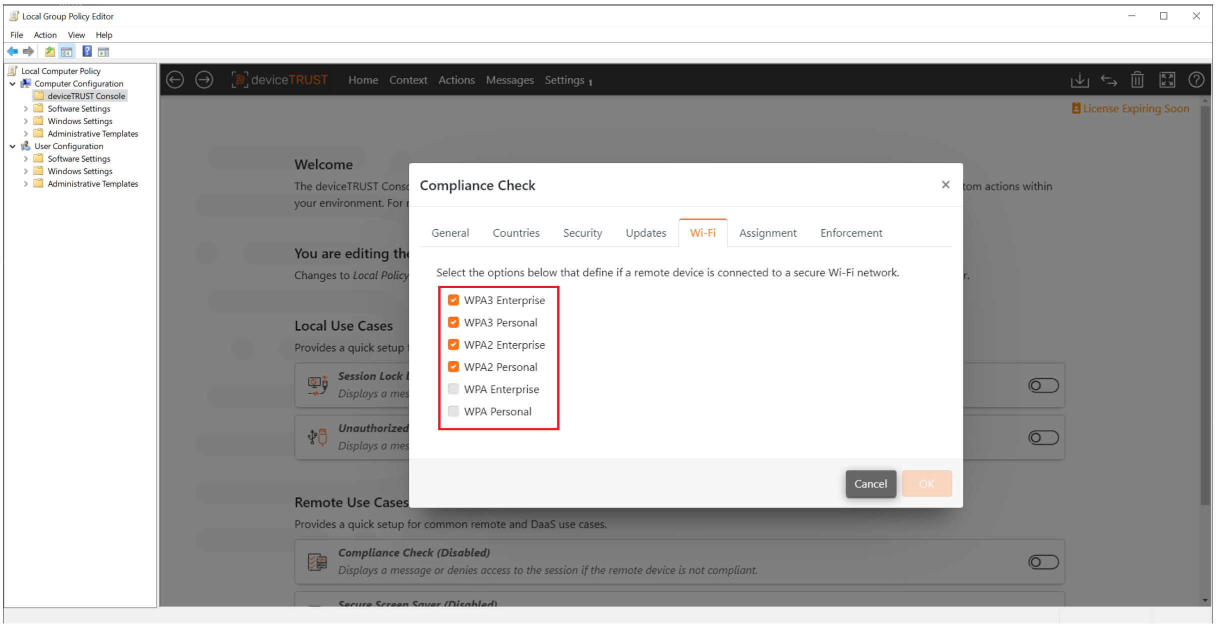
7

days

□ Require IGEL remote devices to be at least version

E.g. 11.06.210

Click on the `WI-FI` configuration tab and enable the `WPA3 ENTERPRISE`, `WPA3 PERSONAL`, `WPA2 ENTERPRISE` and `WPA2 PERSONAL` options.



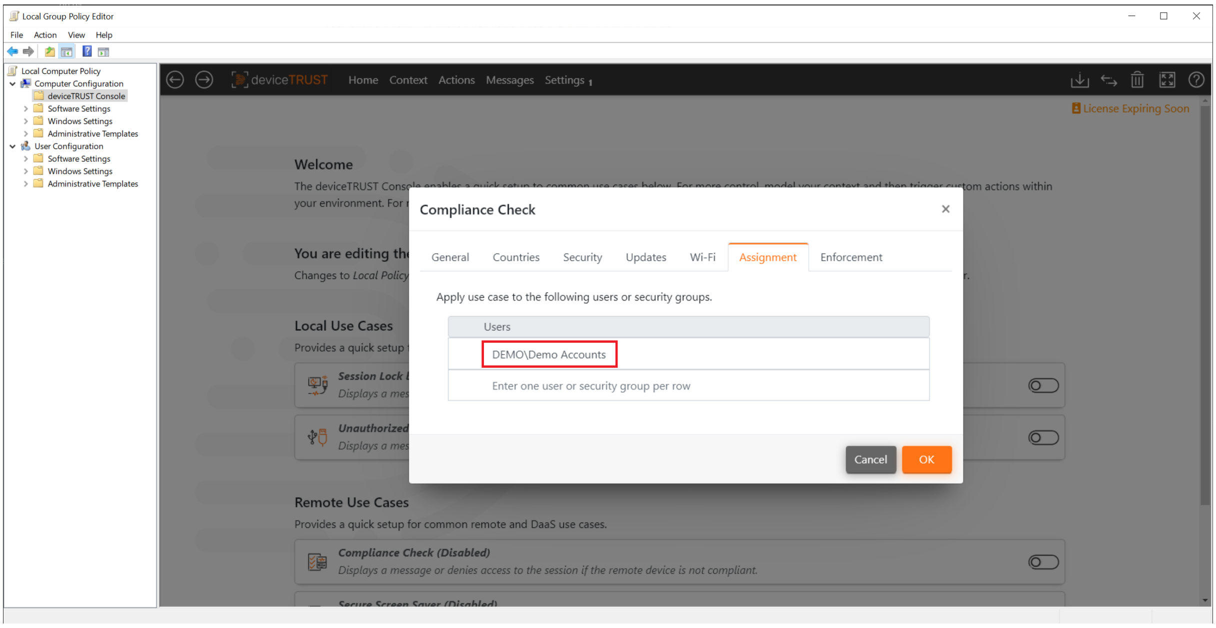
Compliance Check

General | Countries | Security | Updates | Wi-Fi | Assignment | Enforcement

Select the options below that define if a remote device is connected to a secure Wi-Fi network.

- ☑ WPA3 Enterprise
- ☑ WPA3 Personal
- ☑ WPA2 Enterprise
- ☑ WPA2 Personal
- □ WPA Enterprise
- □ WPA Personal

Click on the `ASSIGNMENT` configuration tab and add `USERS` and / or `SECURITY GROUPS` to apply the use case for.



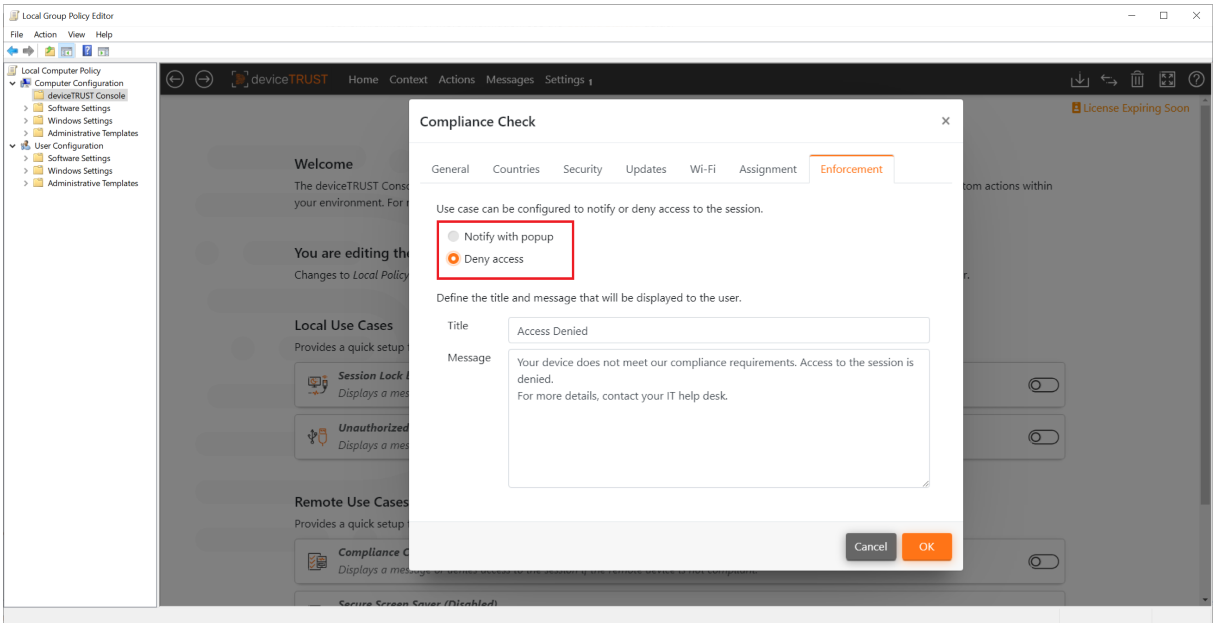
Compliance Check

General | Countries | Security | Updates | Wi-Fi | Assignment | Enforcement

Apply use case to the following users or security groups.

Users
DEMO\Demo Accounts
Enter one user or security group per row

Click on the `ENFORCEMENT` configuration tab and select `DENY ACCESS`.



Compliance Check

General | Countries | Security | Updates | Wi-Fi | Assignment | Enforcement

Use case can be configured to notify or deny access to the session.

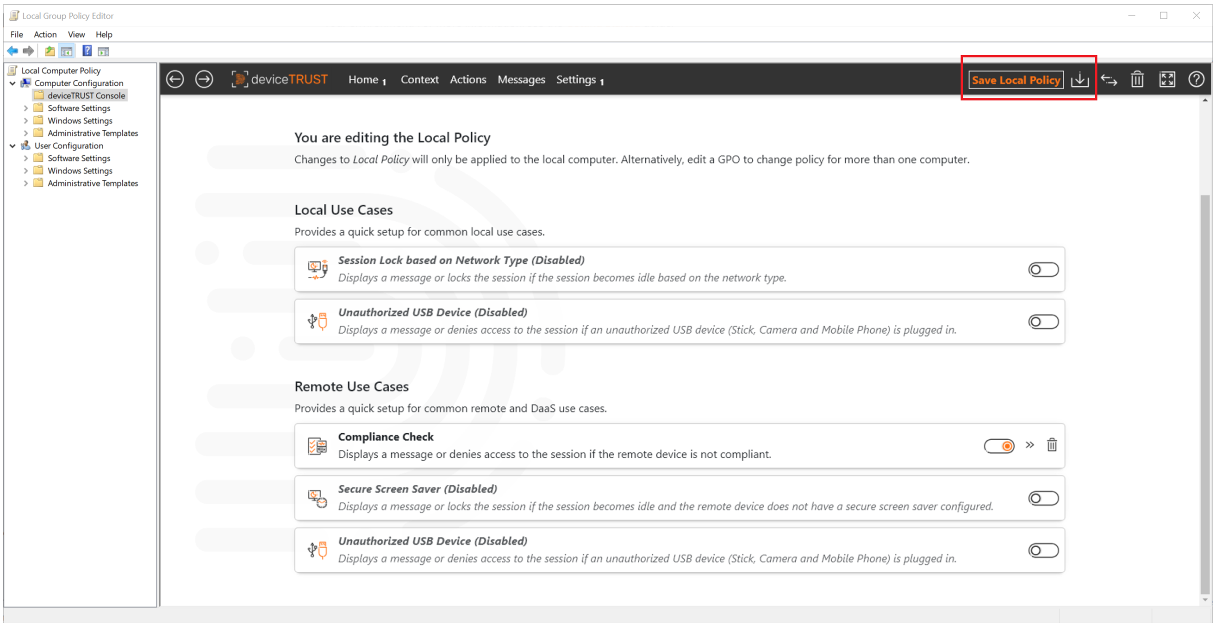
○ Notify with popup
● Deny access

Define the title and message that will be displayed to the user.

Title: Access Denied

Message: Your device does not meet our compliance requirements. Access to the session is denied.
For more details, contact your IT help desk.

Click on `SAVE TO LOCAL COMPUTER POLICY` in the top right toolbar to save the complianc check use case to the local computer policy.



You are editing the Local Policy

Changes to *Local Policy* will only be applied to the local computer. Alternatively, edit a GPO to change policy for more than one computer.

Local Use Cases

Provides a quick setup for common local use cases.

Session Lock based on Network Type (Disabled)
Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

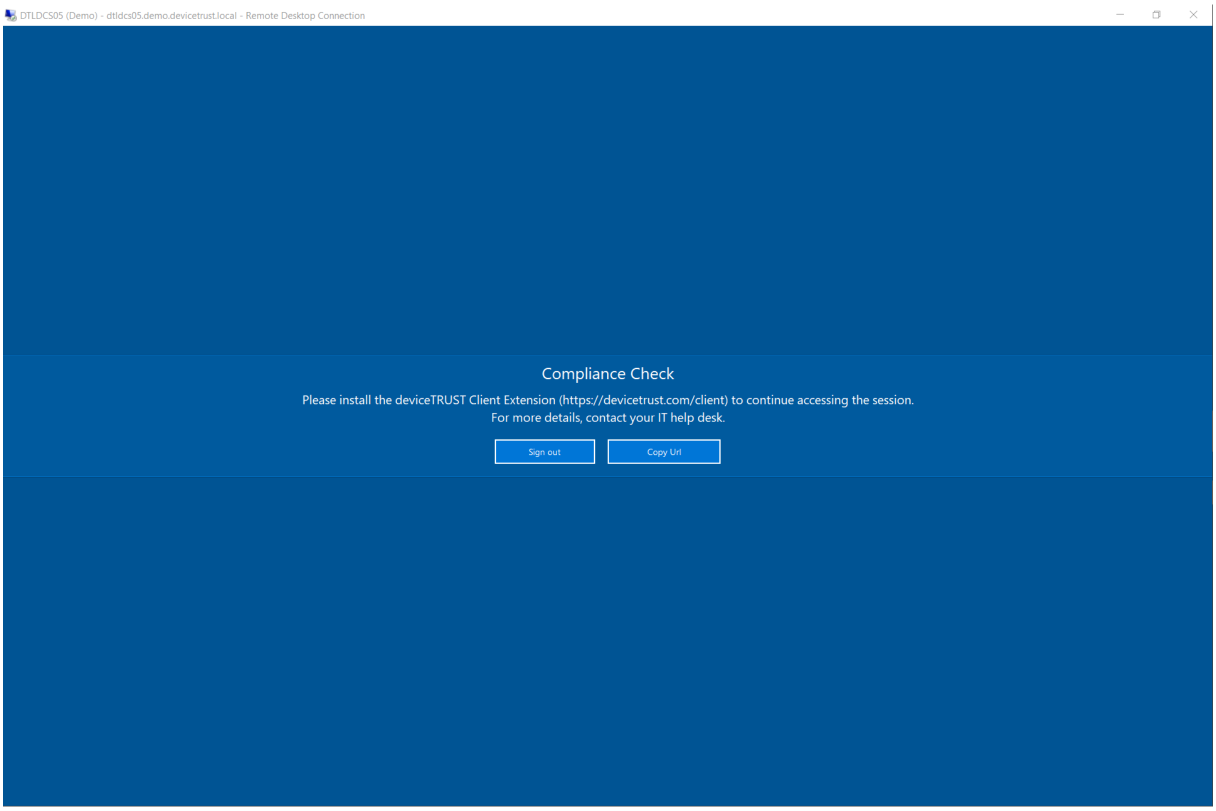
Compliance Check
Displays a message or denies access to the session if the remote device is not compliant.

Secure Screen Saver (Disabled)
Displays a message or locks the session if the session becomes idle and the remote device does not have a secure screen saver configured.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Step 7: Check that access is denied when the deviceTRUST Client Extension is not installed

From a device without the deviceTRUST Client Extension installed, connect to your remoting or DaaS host system. Because the remote device does not have an active deviceTRUST Client Extension, the access will be denied with the following message:



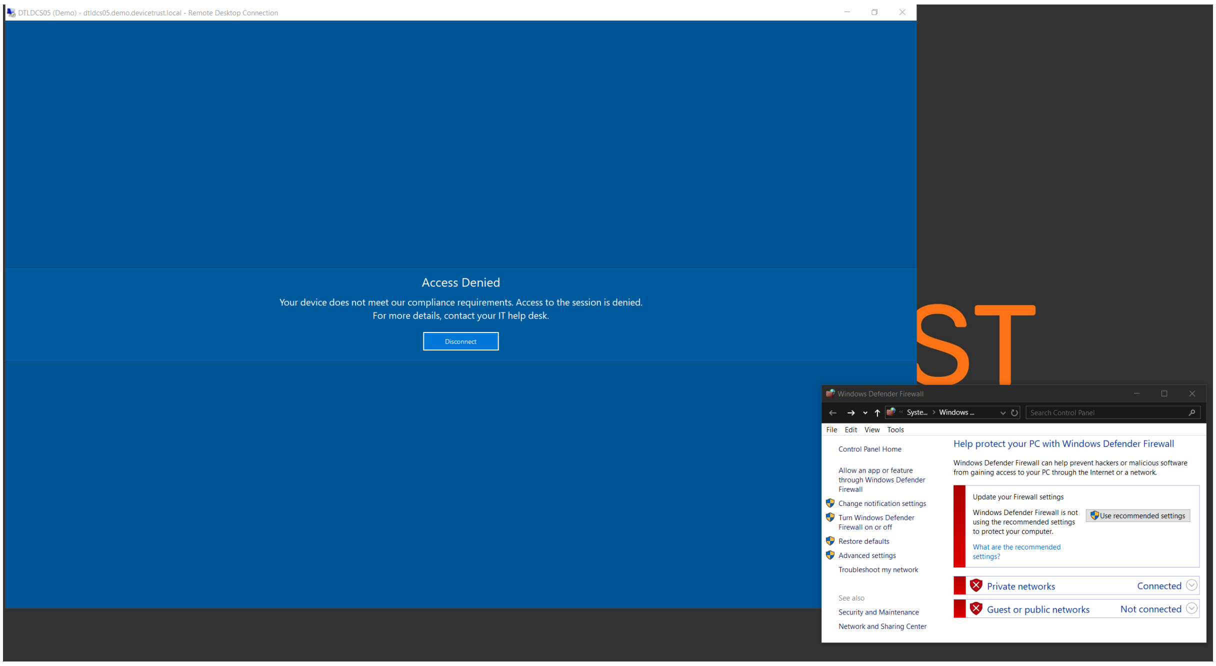
DTLDCS05 (Demo) - dtldcs05.demo.devicetrust.local - Remote Desktop Connection

Compliance Check

Please install the deviceTRUST Client Extension (https://devicetrust.com/client) to continue accessing the session.
For more details, contact your IT help desk.

Step 8: Test the Compliance Check use case from a Microsoft Windows device

From a Microsoft Windows device with the deviceTRUST Client Extension installed, connect to your remoting or DaaS host system. Toggle the state of the Windows Defender Firewall to see how deviceTRUST can simply and dynamically control access to the session depending on the firewall state of the remote device.


DTLDCS05 (Demo) - dtldcs05.demo.devicetrust.local - Remote Desktop Connection
ST
Windows Defender Firewall
File Edit View Tools
Control Panel Home
Allow an app or feature through Windows Defender Firewall
Change notification settings
Turn Windows Defender Firewall on or off
Restore defaults
Advanced settings
Troubleshoot my network
See also
Security and Maintenance
Network and Sharing Center
Help protect your PC with Windows Defender Firewall
Update your Firewall settings
Use recommended settings
What are the recommended settings?
Private networks
Connected
Guest or public networks
Not connected
Disconnect

Access Denied

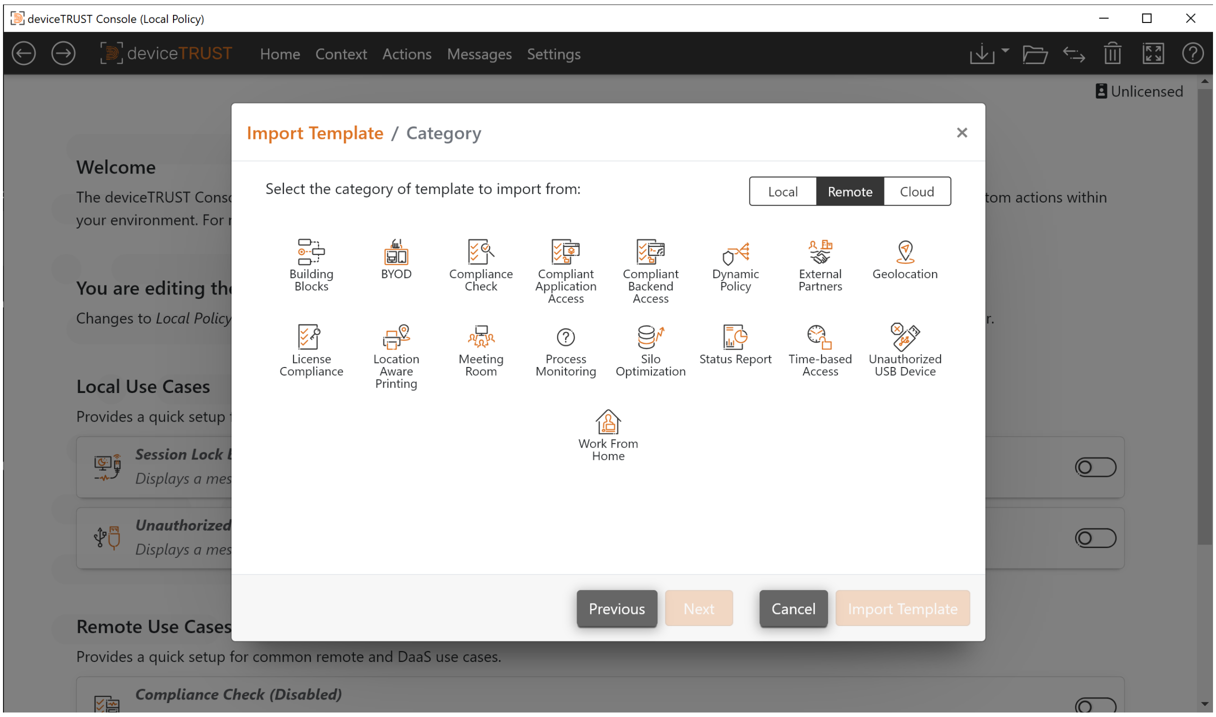
Your device does not meet our compliance requirements. Access to the session is denied.
For more details, contact your IT help desk.

Windows Defender Firewall can help prevent hackers or malicious software from gaining access to your PC through the Internet or a network.

Windows Defender Firewall is not using the recommended settings to protect your computer.

Next steps

You have now successfully implemented your first use case with deviceTRUST for your remoting and DaaS environment. Feel free to check out our additional use cases provided on the deviceTRUST Console homepage under `REMOTE USE CASES`. In addition, the deviceTRUST Console gives you access to many more configuration Templates for a wide variety of use cases.



Select the category of template to import from:

Welcome

The deviceTRUST Cons… tom actions within your environment. For r…

You are editing the…

Changes to *Local Policy* … r.

Local Use Cases

Provides a quick setup …

Displays a mes…

Displays a mes…

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Troubleshooting

If your deviceTRUST installation or configuration does not work as expected, you can use the Troubleshooting guide to start troubleshooting.

Download deviceTRUST 23.1 for Windows

The latest deviceTRUST 23.1 binaries, including the deviceTRUST Agent, Console and Client Extension can be downloaded from the link below:

Version	23.1.410
Release Date	21st October 2024
URL	https://storage.devicetrust.com/download/deviceTRUST-23.1.410.zip
SHA256 Hash	04F5F434178FBE97D25F965A5F29239FF4EB117F04A5140C9D

Previous releases of these components can be downloaded download.

Download deviceTRUST® Client Extensions

The latest client extensions for Microsoft Windows, Apple macOS and Ubuntu can be download by end users from thedeviceTRUST Client Extension Download page.

The latest client extensions for IGEL OS and Unicon™ eLux are built into the respective products.

The latest Stratodesk NoTouch OS client extension can be found below:

Version	23.1.100
Release Date	19th May 2023
NoTouch Citrix URL	https://storage.devicetrust.com/client/dtclient-notouch-amd64-release-23.1.100.0/libdtclient%5Fica.so
NoTouch Citrix SHA256	B5AF15C3338646A62EA4090AEB41C2DD2609B7B9228085C6B
NoTouch RDP URL	https://storage.devicetrust.com/client/dtclient-notouch-amd64-release-23.1.100.0/libdtclient%5Frdp.so
NoTouch RDP SHA256	1E4F5DF9ABBA115BF9E0FD62A7EDA54C46C8C47ABE9EC12B

NoTouch VMware URL	https://storage.devicetrust.com/client/dtclient-notouch-amd64-release-23.1.100.0/libdtclient%5Fvmware.so
NoTouch VMware SHA256	27B80D37F28DC90727560527FD04BDA56F33C919EEE17C9E8

Note:

```
deviceTRUST may change these urls in the future. To ensure you
   don't experience any downtime you should copy these files and
    host them within your own environment.
```

Software components

After downloading the deviceTRUST software, you will find the following components:

Component	Description
DTAGENT-X64-RELEASE-x.x.x.x.MSI	The deviceTRUST Agent installer.
DTCLIENT-EXTENSION-RELEASE-x.x.x.x.EXE	The deviceTRUST Client Extension installer.
DTCONSOLE-X64-RELEASE-x.x.x.x.MSI	The deviceTRUST Console installer.
DTPOLICYDEFINITIONS-x.x.x.x.ZIP	The deviceTRUST ADMX policy definitions for configuring legacy options in the software from Microsoft Active Directory Group Policy (GPO).

Note:

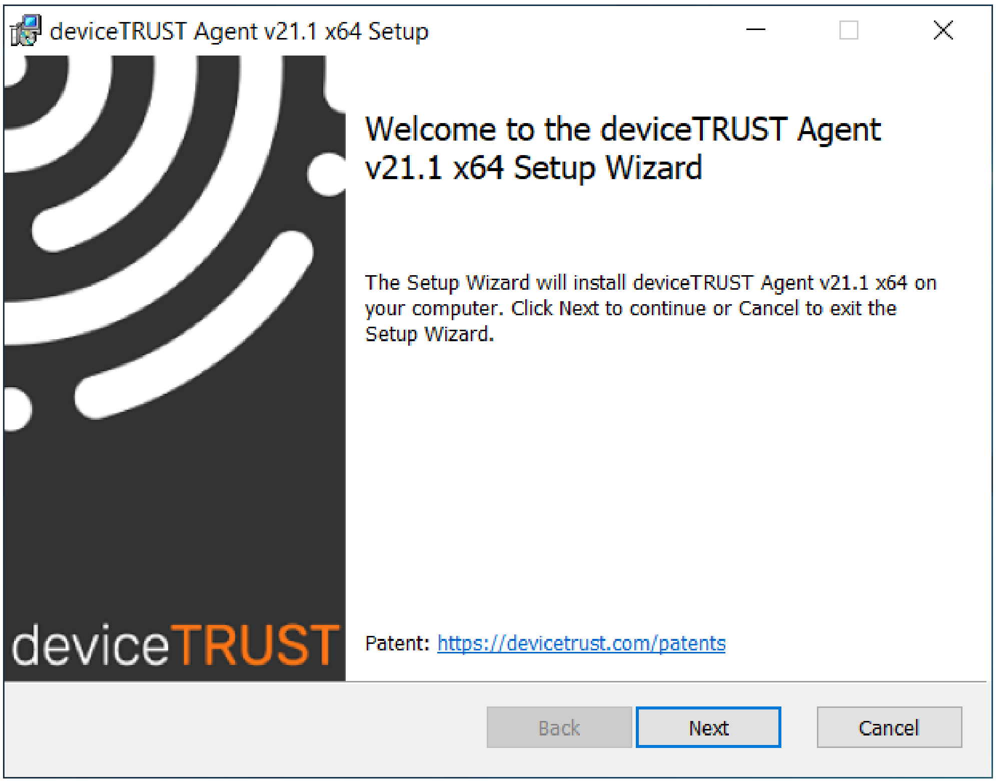
```
deviceTRUST Agent, Console and Client Extension components
   require administrative privileges for the installation.
```

Installation

TABLE OF CONTENTS

Installing the deviceTRUST® Agent

The deviceTRUST Agent requires a user account with local administrative privileges to install the deviceTRUST Agent on the target system. The installation can be performed by following the steps of the deviceTRUST Agent installer.

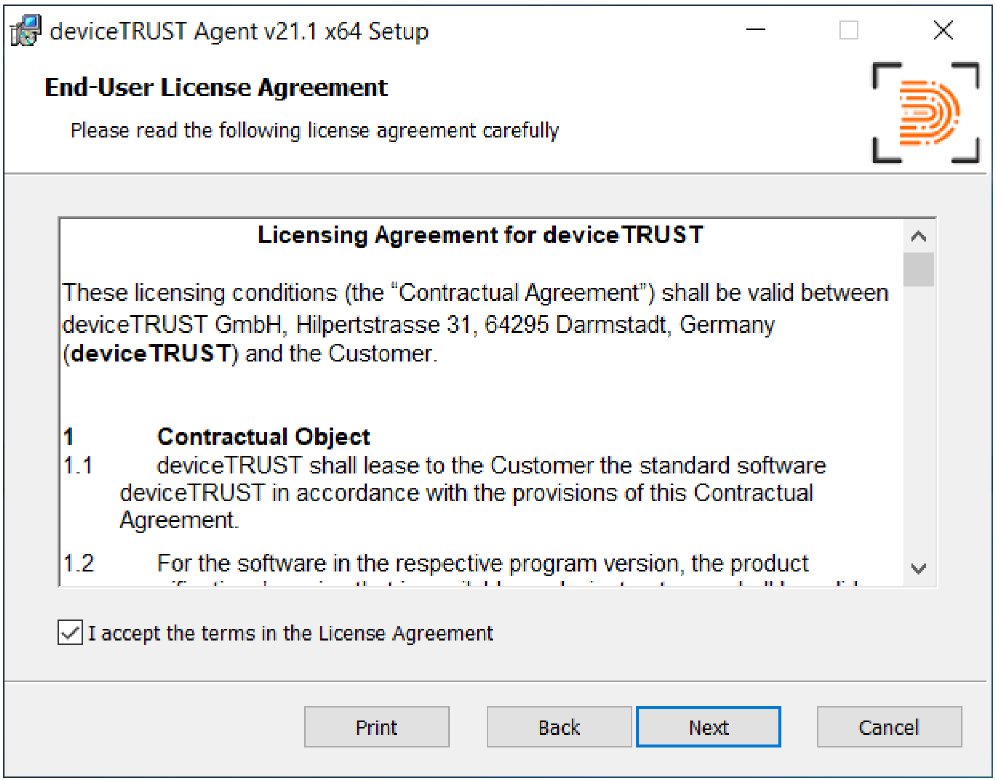


deviceTRUST Agent v21.1 x64 Setup

Welcome to the deviceTRUST Agent v21.1 x64 Setup Wizard

The Setup Wizard will install deviceTRUST Agent v21.1 x64 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Patent: https://devicetrust.com/patents



deviceTRUST Agent v21.1 x64 Setup

End-User License Agreement

Please read the following license agreement carefully

Licensing Agreement for deviceTRUST

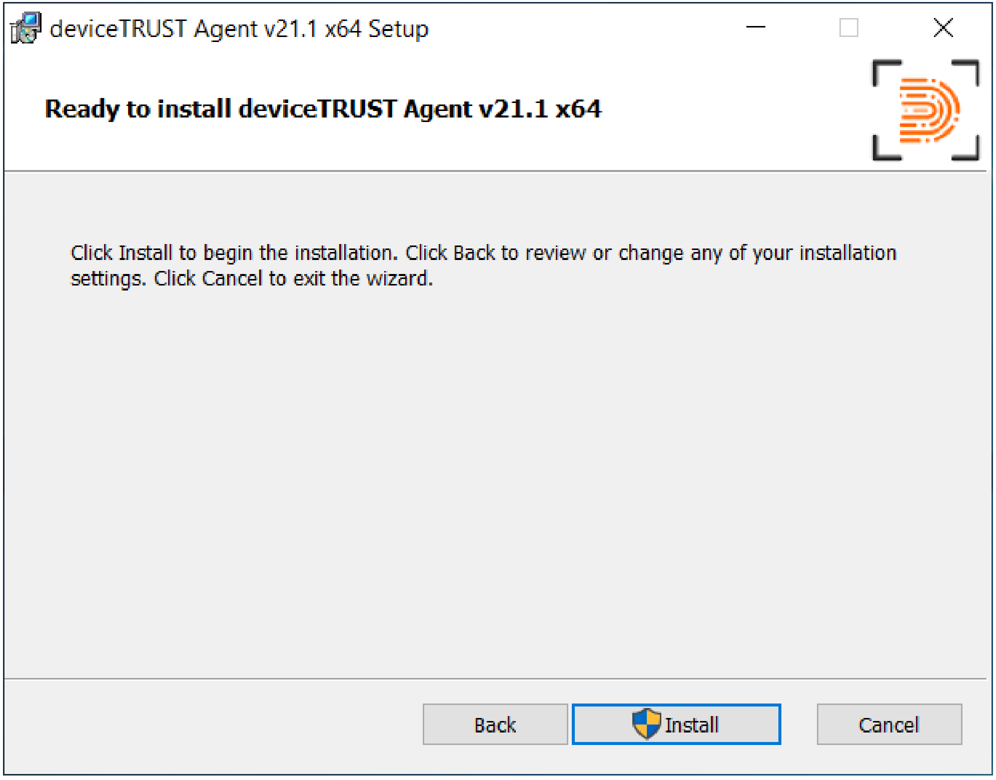
These licensing conditions (the "Contractual Agreement") shall be valid between deviceTRUST GmbH, Hilpertstrasse 31, 64295 Darmstadt, Germany (**deviceTRUST**) and the Customer.

1 Contractual Object

1.1 deviceTRUST shall lease to the Customer the standard software deviceTRUST in accordance with the provisions of this Contractual Agreement.

1.2 For the software in the respective program version, the product

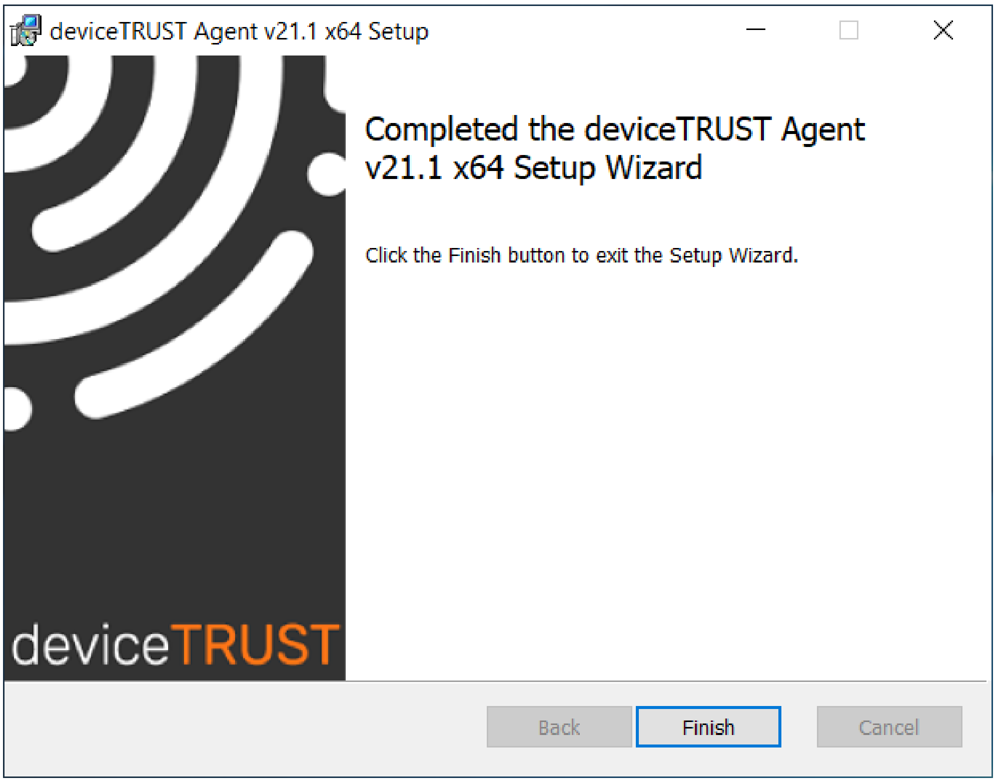
☑ I accept the terms in the License Agreement



deviceTRUST Agent v21.1 x64 Setup

Ready to install deviceTRUST Agent v21.1 x64

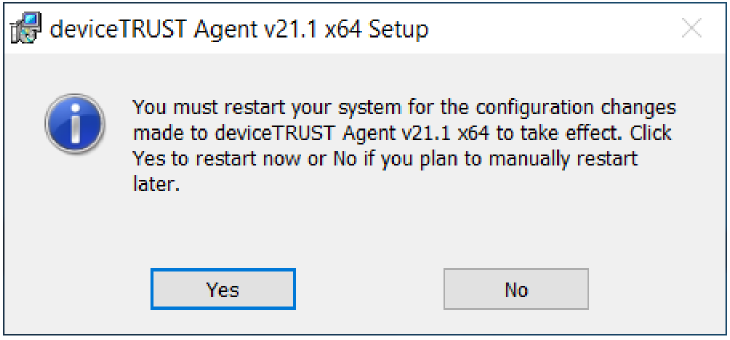
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.



deviceTRUST Agent v21.1 x64 Setup

Completed the deviceTRUST Agent v21.1 x64 Setup Wizard

Click the Finish button to exit the Setup Wizard.



deviceTRUST Agent v21.1 x64 Setup

You must restart your system for the configuration changes made to deviceTRUST Agent v21.1 x64 to take effect. Click Yes to restart now or No if you plan to manually restart later.

> **Note:**
>
> Installation path: %PROGRAMFILES%\DEVICETRUST\AGENT
>
> If the installation of the deviceTRUST Agent has finished successfully, a reboot is required to enable deviceTRUST to get system notifications to act on.
>
> If the *Remote Desktop Services* server role is added after installing the deviceTRUST Agent, the deviceTRUST Agent will need to be reinstalled.
>
> The deviceTRUST Agent will not function until a valid license is applied.

Citrix Virtual Channel Security

When using Citrix Virtual Apps and Desktops, you may need to edit the `Virtual channel allow list` policy to allow the deviceTRUST Agent to open a virtual channel to the deviceTRUST Client.

More details can be found on the Knowledge Base.

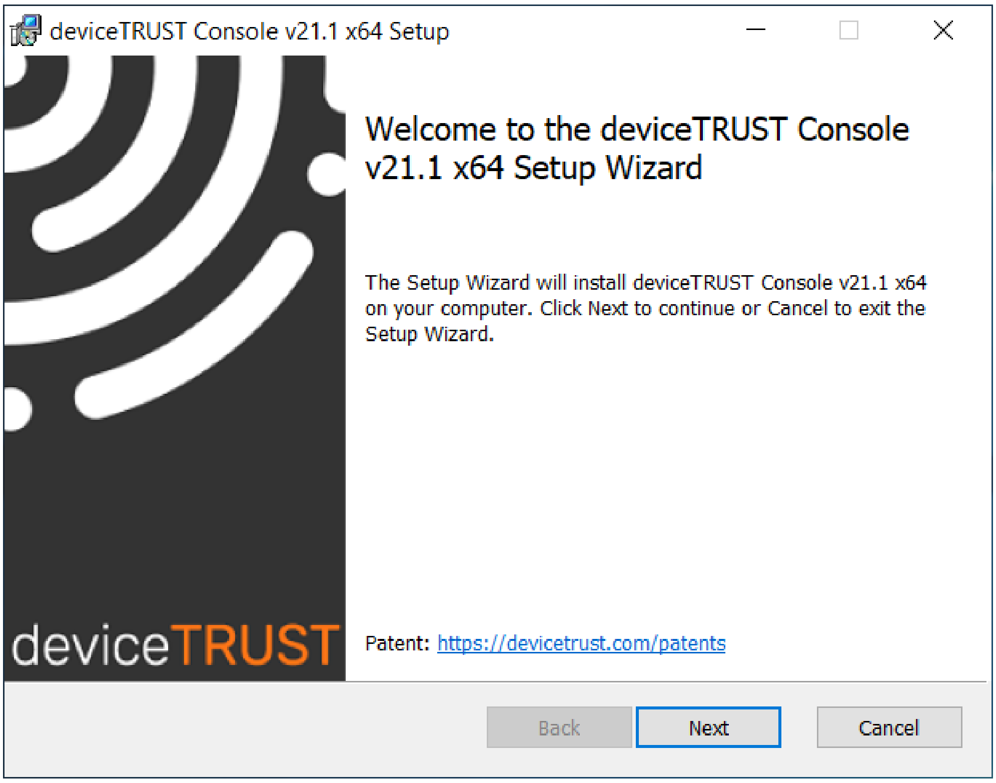
Unattended Installation

The deviceTRUST Agent can be installed unattended from the command line interface with the following options:

Component	Commandline
dtagent-x64-release-x.x.x.x.msi	The deviceTRUST Agent installer file can be customized by common Microsoft Windows Installer parameters. An unattended installation can be achieved with the following parameters `MSIEXEC.EXE /I DTAGENT-X64-RELEASE-X.X.X.X.MSI /PASSIVE / FORCERESTART`

Installing the deviceTRUST® Console

The deviceTRUST Console requires a user account with local administrative privileges to install the deviceTRUST Console on the targeting system. The installation can be performed by following the steps of the deviceTRUST Console installer.

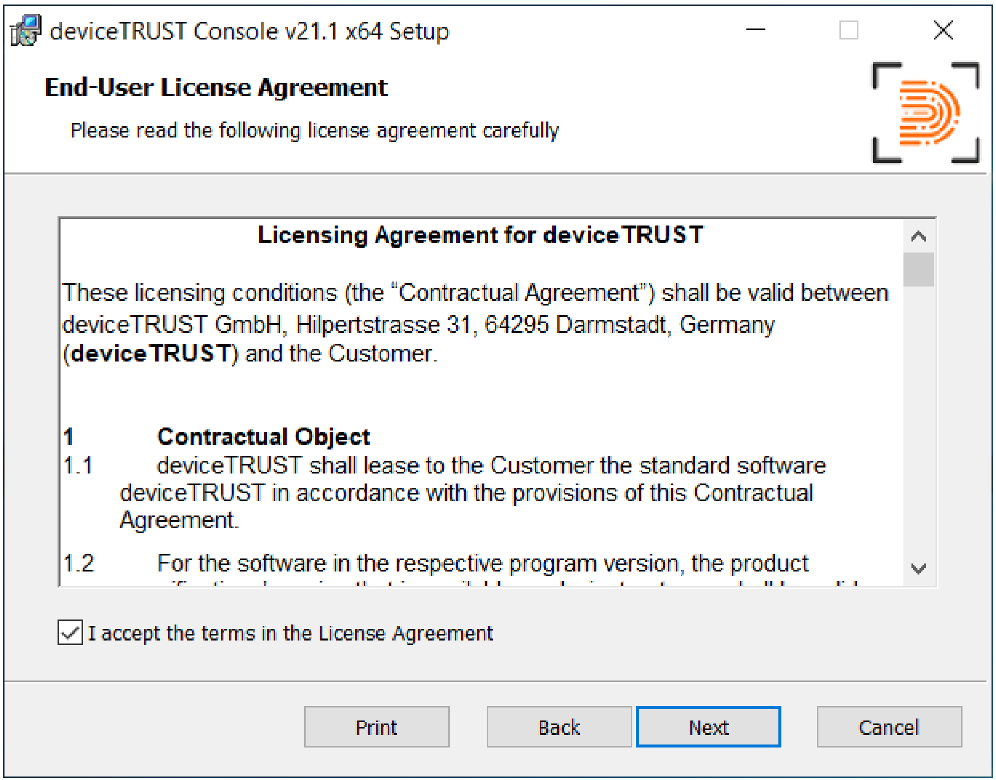


deviceTRUST Console v21.1 x64 Setup

Welcome to the deviceTRUST Console v21.1 x64 Setup Wizard

The Setup Wizard will install deviceTRUST Console v21.1 x64 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Patent: https://devicetrust.com/patents



deviceTRUST Console v21.1 x64 Setup

End-User License Agreement

Please read the following license agreement carefully

Licensing Agreement for deviceTRUST

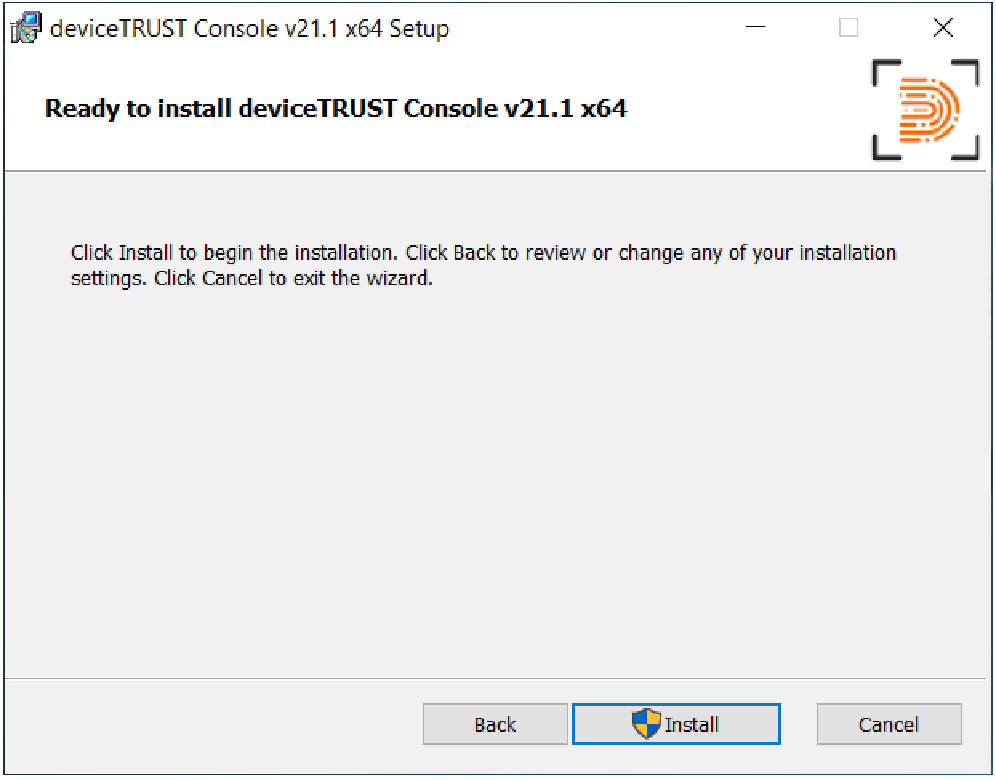
These licensing conditions (the "Contractual Agreement") shall be valid between deviceTRUST GmbH, Hilpertstrasse 31, 64295 Darmstadt, Germany (**deviceTRUST**) and the Customer.

1 Contractual Object

1.1 deviceTRUST shall lease to the Customer the standard software deviceTRUST in accordance with the provisions of this Contractual Agreement.

1.2 For the software in the respective program version, the product

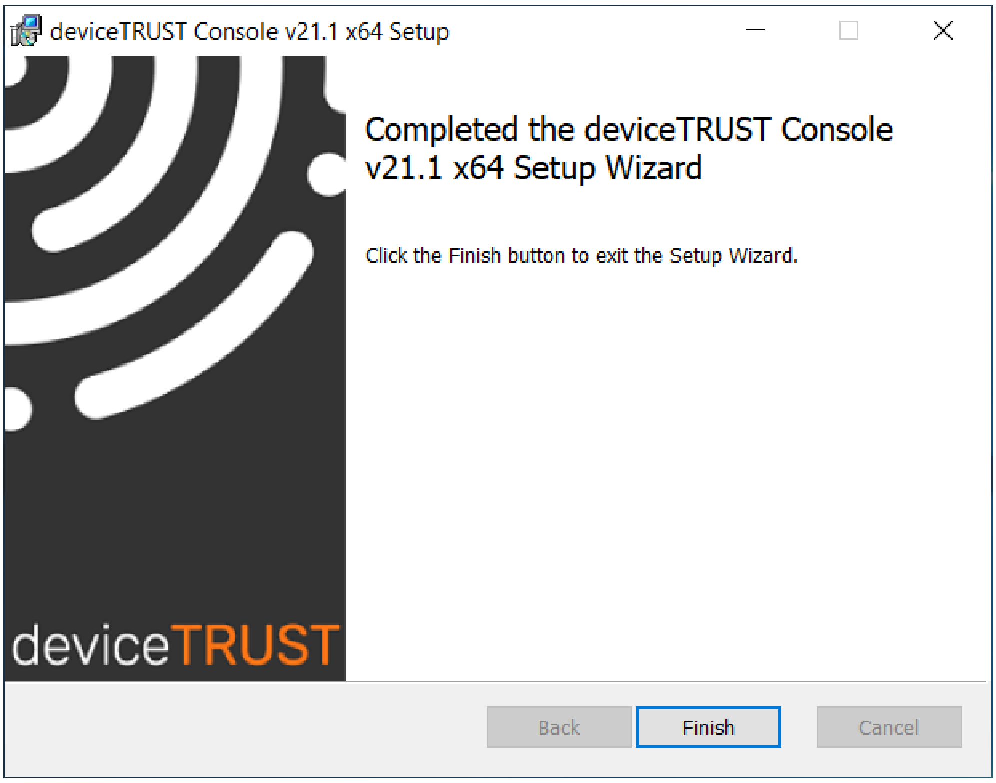
☑ I accept the terms in the License Agreement



deviceTRUST Console v21.1 x64 Setup

Ready to install deviceTRUST Console v21.1 x64

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.



deviceTRUST Console v21.1 x64 Setup

Completed the deviceTRUST Console v21.1 x64 Setup Wizard

Click the Finish button to exit the Setup Wizard.

> **Note**
>
> Installation path: `%PROGRAMFILES%\DEVICETRUST\CONSOLE`

Unattended Installation

The deviceTRUST Console can be installed unattended from the command line interface with the following options:

Component	Commandline
dtconsole-x64-release-x.x.x.x.msi	The deviceTRUST Console installer file can be customized by common Microsoft Windows Installer parameters. An unattended installation can be achieved with the following parameters `MSIEXEC.EXE /I DTCONSOLE-X64-RELEASE-X.X.X.X.MSI /PASSIVE`

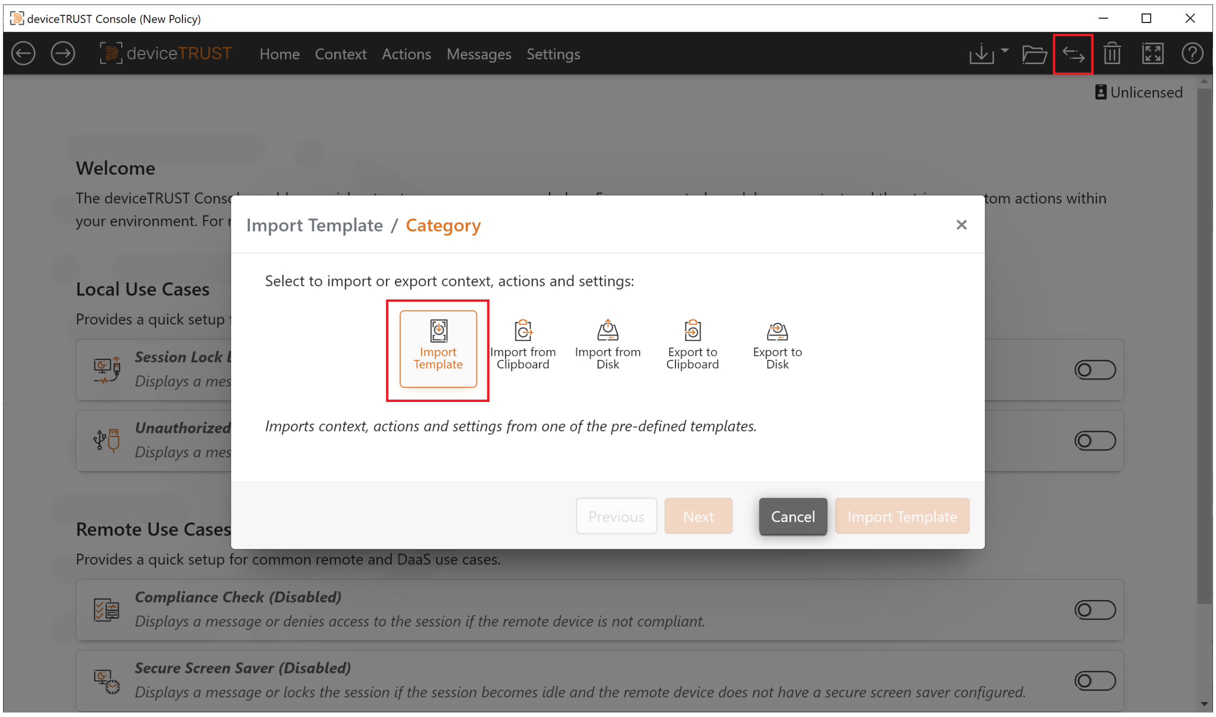
Installing the deviceTRUST® Client Extension

TABLE OF CONTENTS

- Apple macOS
- Apple iOS and iPadOS
- Ubuntu
- IGEL OS
- Stratodesk NoTouch OS
- Unicon eLux RP6

Templates

The deviceTRUST® Console includes a set of templates which can be used to quickly implement a use case. Launch the deviceTRUST Console and select **SHARING** in the top right of the navigation bar, then choose **IMPORT TEMPLATE**.



Welcome

The deviceTRUST Cons… tom actions within your environment. For r…

Local Use Cases

Provides a quick setup …

Session Lock …

Displays a mes…

Unauthorized…

Displays a mes…

Import Template / Category

Select to import or export context, actions and settings:

Imports context, actions and settings from one of the pre-defined templates.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)

Displays a message or denies access to the session if the remote device is not compliant.

Secure Screen Saver (Disabled)

Displays a message or locks the session if the session becomes idle and the remote device does not have a secure screen saver configured.

The deviceTRUST use cases are summarized in the following categories for each target platform. Use the filter to select the desired target platform.

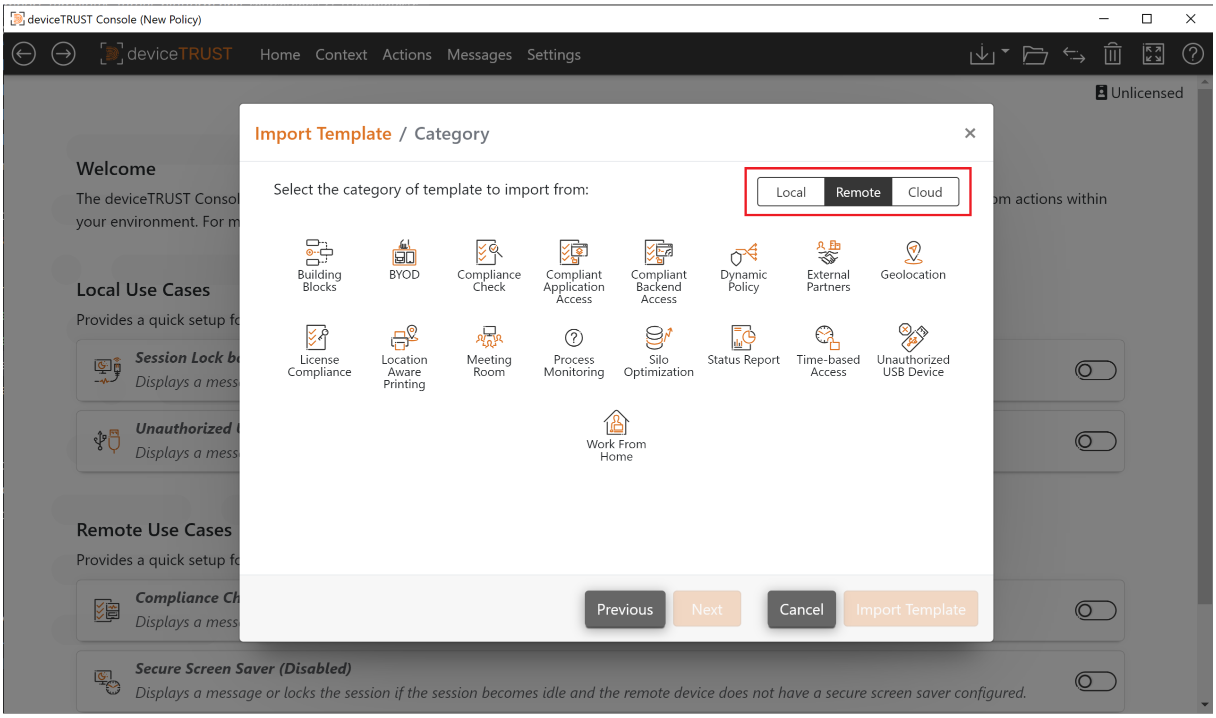
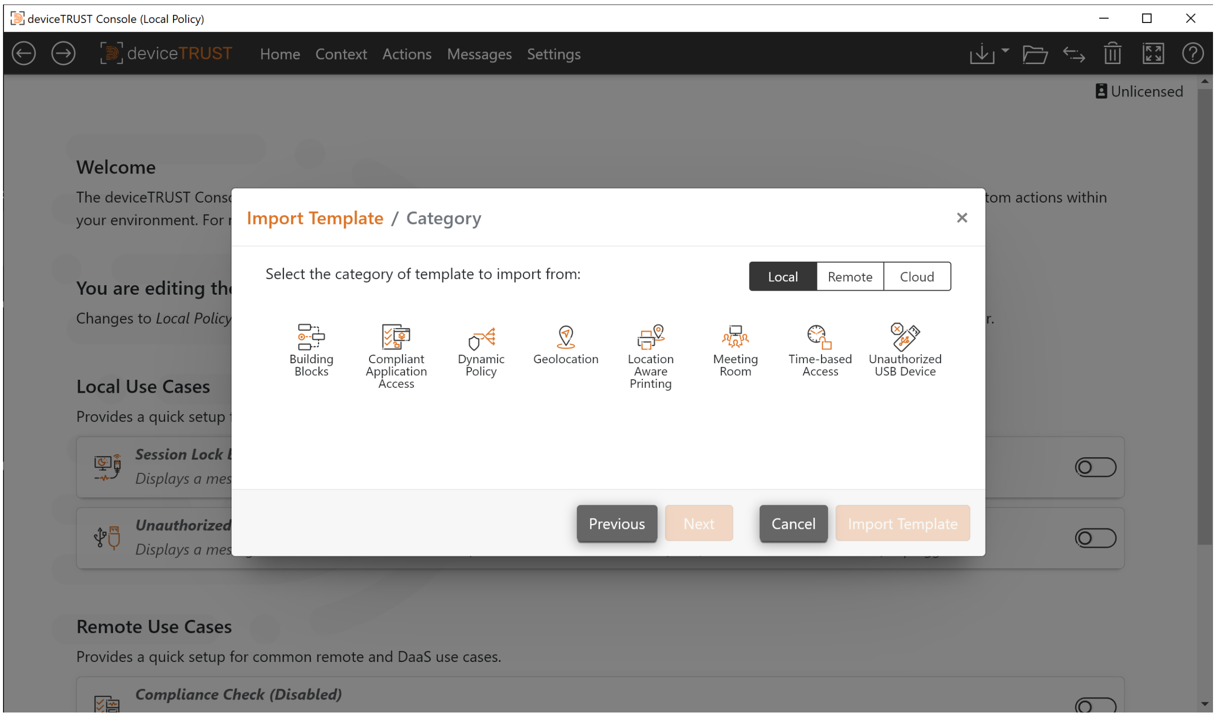


Table of Contents

Local Templates

The deviceTRUST® use cases for local devices are summarized within the following categories.



Welcome

The deviceTRUST Cons… tom actions within your environment. For …

You are editing th…

Changes to *Local Policy* … r.

Local Use Cases

Provides a quick setup …

Session Lock …

Displays a mes…

Unauthorized…

Displays a mes…

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)

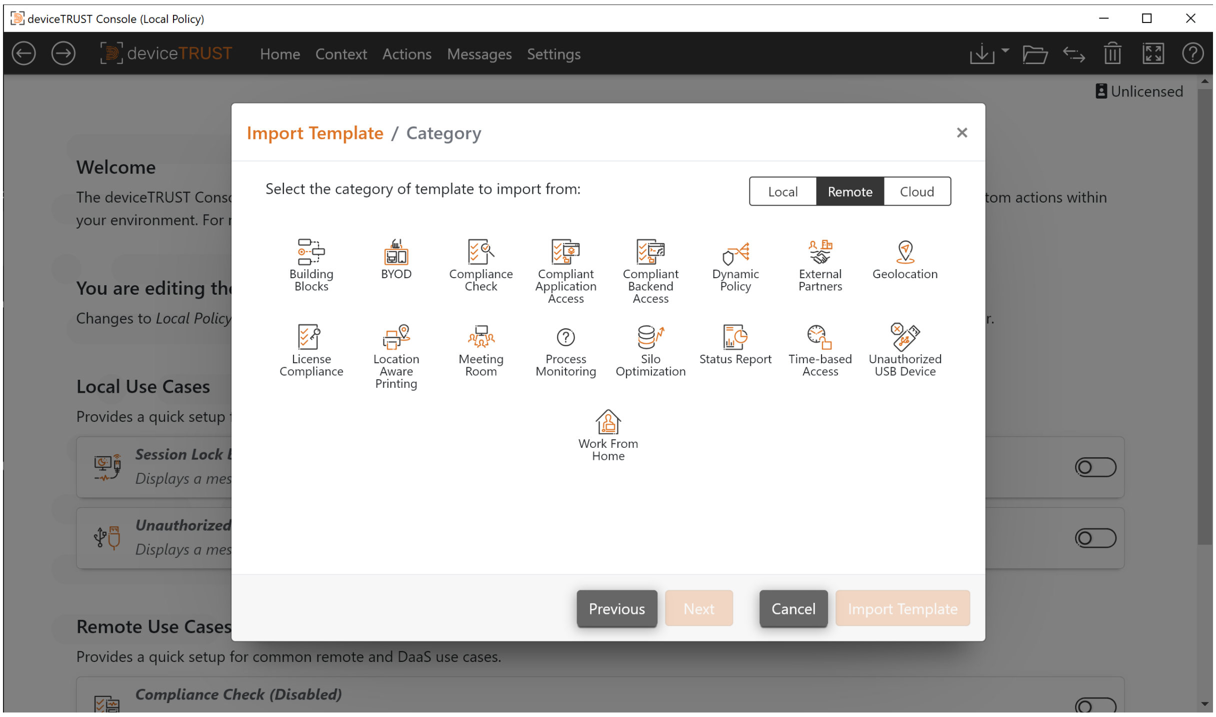
Import Template / Category

Select the category of template to import from:

Table of Contents

Remote Templates

The deviceTRUST® use cases for remoting and DaaS are summarized within the following categories.



Select the category of template to import from:

The deviceTRUST Cons… tom actions within your environment. For i…

Changes to *Local Policy* … r.

Provides a quick setup f…

Displays a mes…

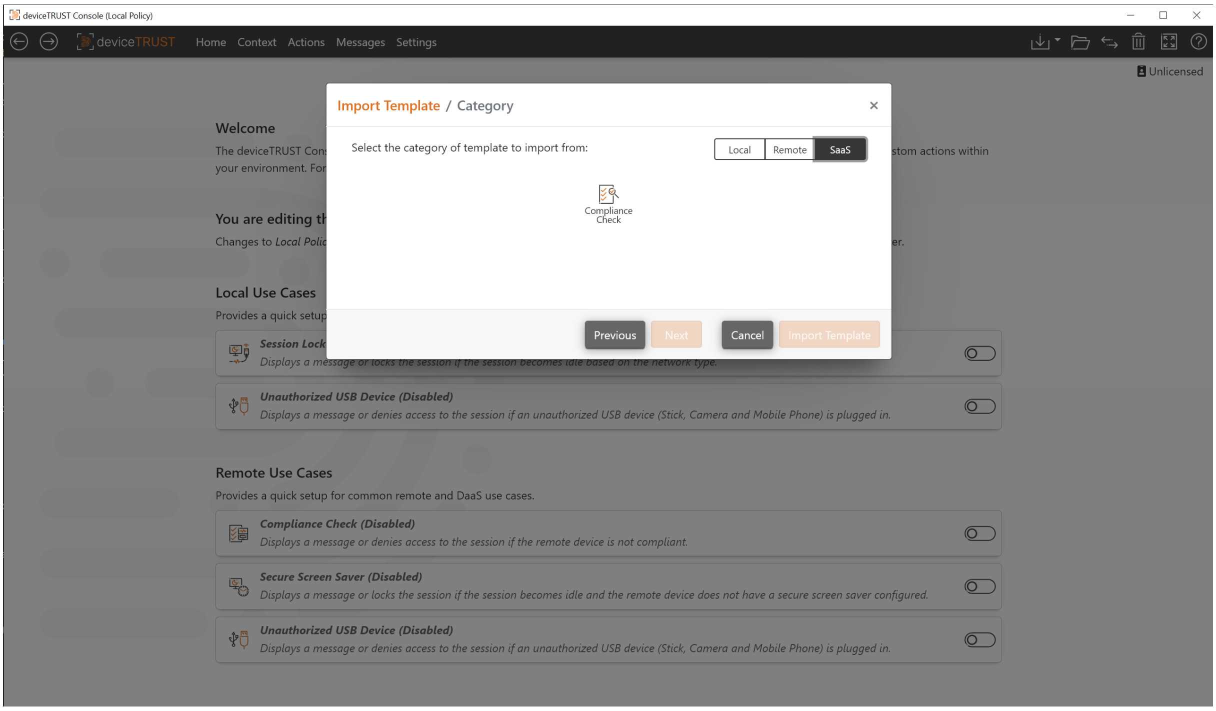
Displays a mes…

Provides a quick setup for common remote and DaaS use cases.

Table of Contents

SaaS Templates

The deviceTRUST® use cases for SaaS connected devices are summarized within the following categories.



Select the category of template to import from:

Welcome

The deviceTRUST Con… stom actions within your environment. For…

You are editing th…

Changes to *Local Polic*… er.

Local Use Cases

Provides a quick setup…

Session Lock…
Displays a message or locks the session if the session becomes idle based on the network type.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Remote Use Cases

Provides a quick setup for common remote and DaaS use cases.

Compliance Check (Disabled)
Displays a message or denies access to the session if the remote device is not compliant.

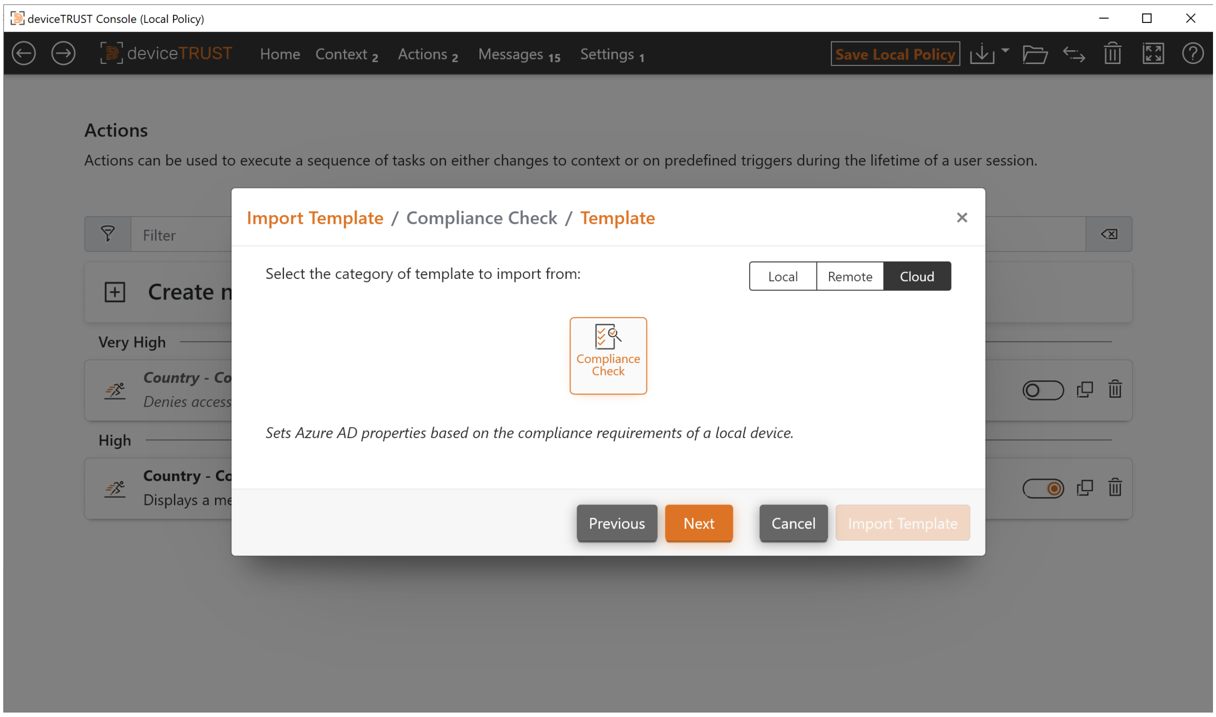
Secure Screen Saver (Disabled)
Displays a message or locks the session if the session becomes idle and the remote device does not have a secure screen saver configured.

Unauthorized USB Device (Disabled)
Displays a message or denies access to the session if an unauthorized USB device (Stick, Camera and Mobile Phone) is plugged in.

Table of Contents

Compliance Check

Sets Microsoft Entra ID properties based on the compliance requirements of a local device.



Actions

Actions can be used to execute a sequence of tasks on either changes to context or on predefined triggers during the lifetime of a user session.

Select the category of template to import from:

Sets Azure AD properties based on the compliance requirements of a local device.

TABLE OF CONTENTS

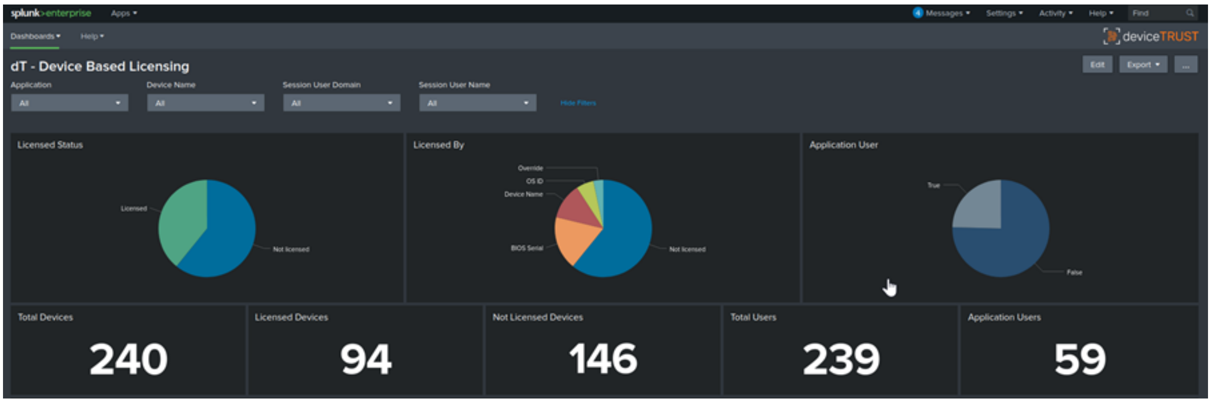
Reporting

Table of Contents

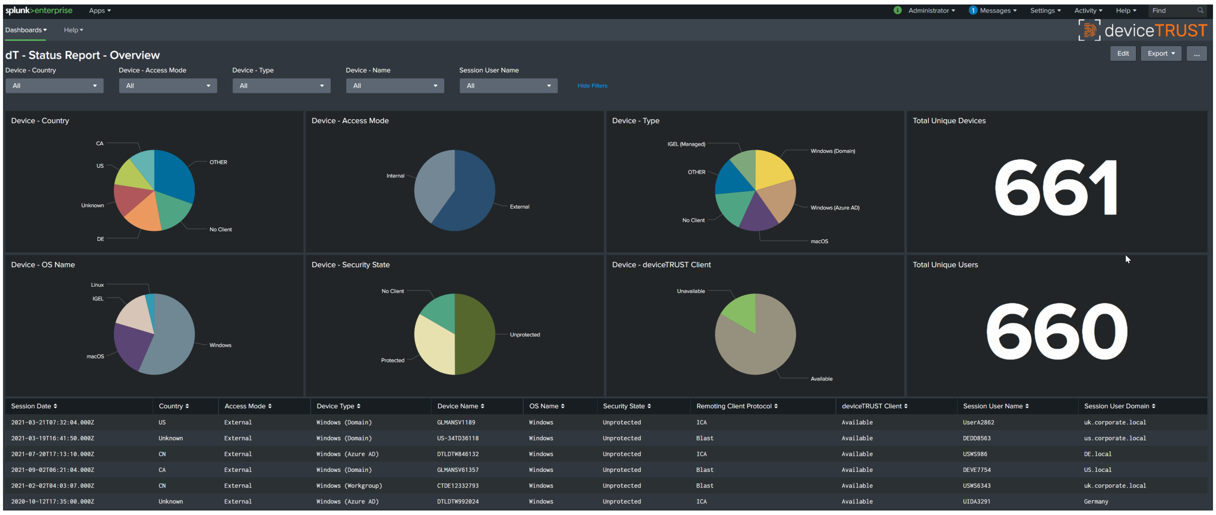
Splunk Dashboards

deviceTRUST® includes a Splunk app to easily create a Splunk dashboard to monitor the contextual status of your remoting and DaaS environment.

The License Compliance Templates can be used with the Splunk app to monitor or enforce Device-Based Licensing requirements for one or more applications:



The Splunk Status Report Template can be used to monitor the status of your remoting and DaaS environment:



Session Date ⬍	Country ⬍	Access Mode ⬍	Device Type ⬍	Device Name ⬍	OS Name ⬍	Security State ⬍	Remoting Client Protocol ⬍	deviceTRUST Client ⬍	Session User Name ⬍	Session User Domain ⬍
2021-03-21T07:32:04.000Z	US	External	Windows (Domain)	GLMANSV1189	Windows	Unprotected	ICA	Available	UserA2862	uk.corporate.local
2021-03-19T16:41:50.000Z	Unknown	External	Windows (Domain)	US-34TD36118	Windows	Unprotected	Blast	Available	DEDD8563	us.corporate.local
2021-07-20T17:13:10.000Z	CN	External	Windows (Azure AD)	DTLDTW846132	Windows	Unprotected	ICA	Available	USWS986	DE.local
2021-09-02T06:21:04.000Z	CA	External	Windows (Domain)	GLMANSV61357	Windows	Unprotected	Blast	Available	DEVE7754	US.local
2021-02-02T04:03:07.000Z	CN	External	Windows (Workgroup)	CTDE12332793	Windows	Unprotected	Blast	Available	USWS6343	uk.corporate.local
2020-10-12T17:35:00.000Z	Unknown	External	Windows (Azure AD)	DTLDTW992024	Windows	Unprotected	ICA	Available	UIDA3291	Germany

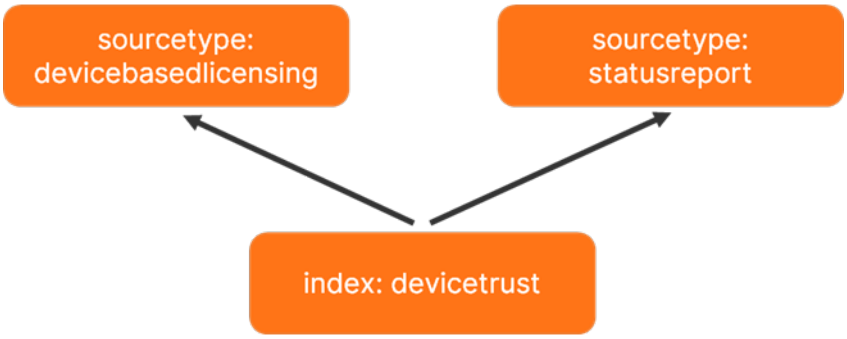
The following steps will be performed:

Step 1: Creating the Splunk Index

Splunk collects data in *indexes* which holds the log data and make it searchable. There are different methods for working with indexes. Data can, for example, flow into one common index. Alternatively, multiple indexes can be created, one for each use case or scenario.

The deviceTRUST reports use one common index `devicetrust` for storing the data. The separation for the different uses cases is done by applying *sourcetypes* `devicebasedlicensing` and `statusreport`.

Our deviceTRUST reports are built on the described combination of index and sourcetypes. If your implementation differs, the reports will have to be adjusted accordingly. Make sure to let us know, we'll happily assist.



sourcetype: devicebasedlicensing

sourcetype: statusreport

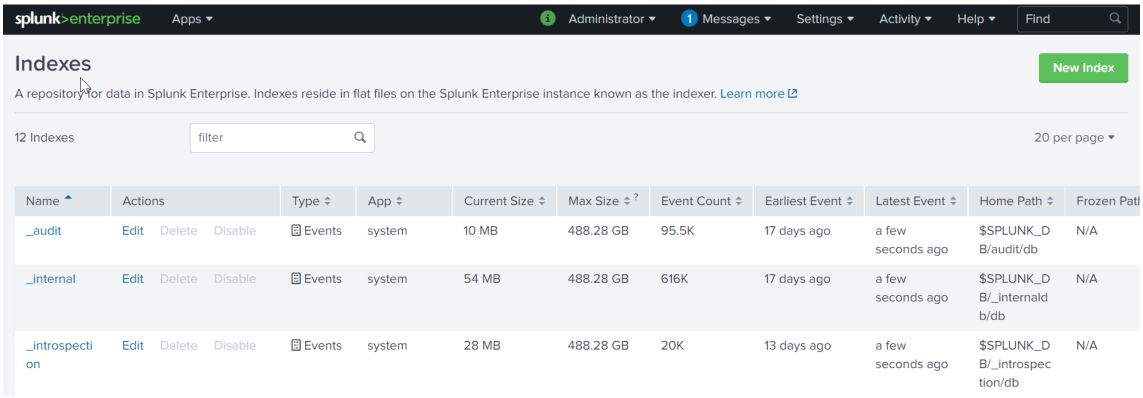
index: devicetrust

You can either create the index `devicetrust` manually or by importing our prepared app `dt_index`.

Step 1.1: Manually creating the Splunk Index

Implementing the index manually does not require any special configuration.

- Open your Splunk management GUI
- Navigate to `Settings\Indexes` and click `New Index`.



Indexes

A repository for data in Splunk Enterprise. Indexes reside in flat files on the Splunk Enterprise instance known as the indexer. Learn more

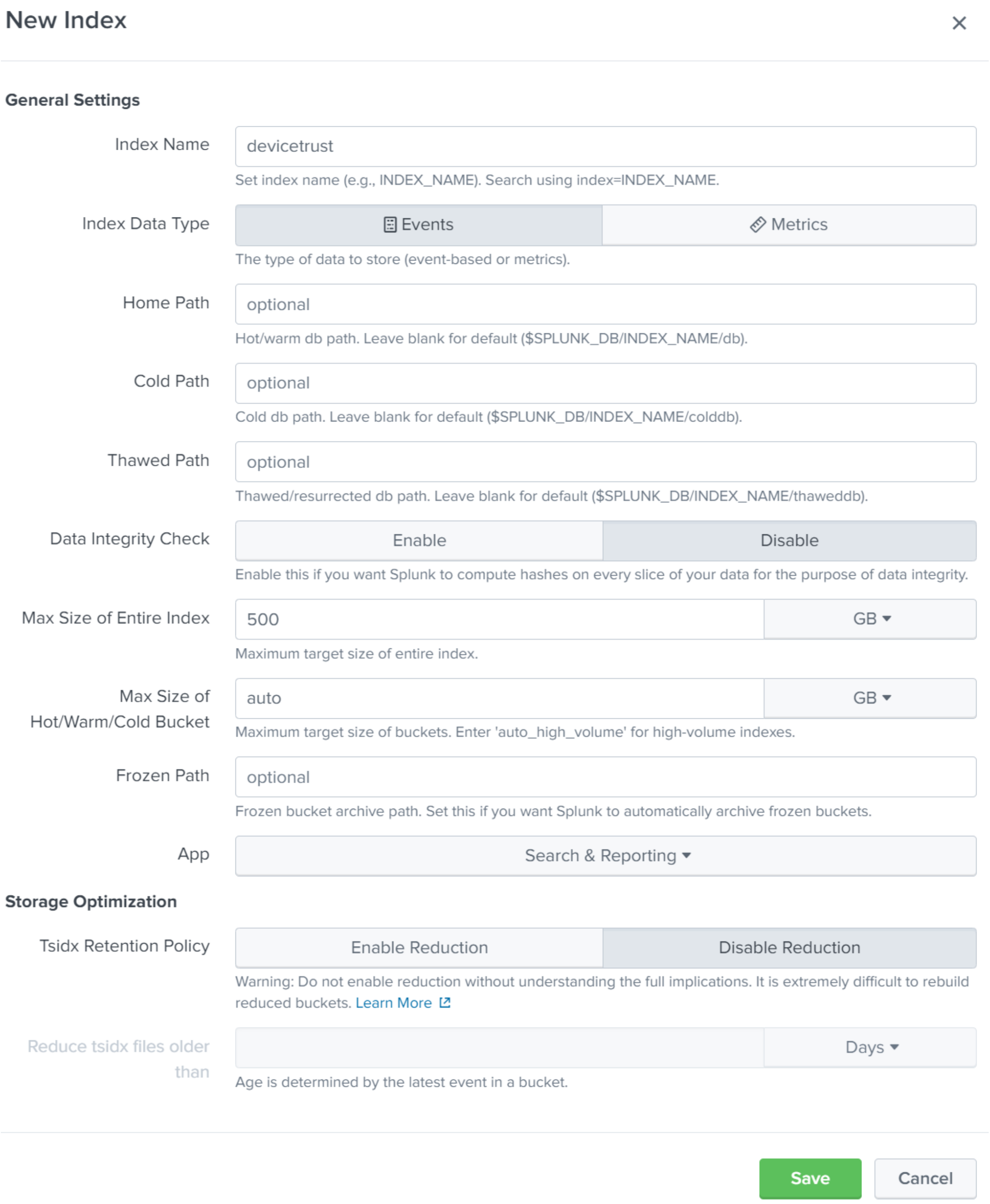
12 Indexes

filter

20 per page

Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen Pat
_audit	Edit Delete Disable	Events	system	10 MB	488.28 GB	95.5K	17 days ago	a few seconds ago	$SPLUNK_DB/audit/db	N/A
_internal	Edit Delete Disable	Events	system	54 MB	488.28 GB	616K	17 days ago	a few seconds ago	$SPLUNK_DB/_internaldb/db	N/A
_introspection	Edit Delete Disable	Events	system	28 MB	488.28 GB	20K	13 days ago	a few seconds ago	$SPLUNK_DB/_introspection/db	N/A

- Set the name to `devicetrust` and all other options are optional.



New Index

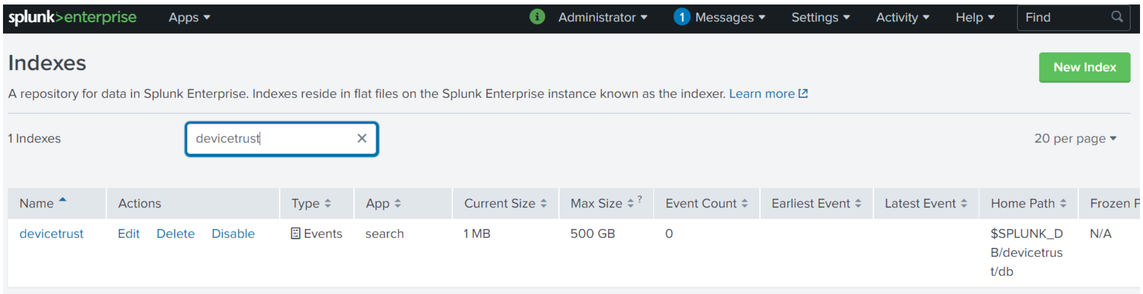
General Settings

Index Name	devicetrust Set index name (e.g., INDEX_NAME). Search using index=INDEX_NAME.
Index Data Type	Events / Metrics The type of data to store (event-based or metrics).
Home Path	optional Hot/warm db path. Leave blank for default ($SPLUNK_DB/INDEX_NAME/db).
Cold Path	optional Cold db path. Leave blank for default ($SPLUNK_DB/INDEX_NAME/colddb).
Thawed Path	optional Thawed/resurrected db path. Leave blank for default ($SPLUNK_DB/INDEX_NAME/thaweddb).
Data Integrity Check	Enable / Disable Enable this if you want Splunk to compute hashes on every slice of your data for the purpose of data integrity.
Max Size of Entire Index	500 GB Maximum target size of entire index.
Max Size of Hot/Warm/Cold Bucket	auto GB Maximum target size of buckets. Enter 'auto_high_volume' for high-volume indexes.
Frozen Path	optional Frozen bucket archive path. Set this if you want Splunk to automatically archive frozen buckets.
App	Search & Reporting

Storage Optimization

Tsidx Retention Policy	Enable Reduction / Disable Reduction Warning: Do not enable reduction without understanding the full implications. It is extremely difficult to rebuild reduced buckets. Learn More
Reduce tsidx files older than	Days Age is determined by the latest event in a bucket.

- Click `Save` and the index will be created.



Indexes

A repository for data in Splunk Enterprise. Indexes reside in flat files on the Splunk Enterprise instance known as the indexer. Learn more

1 Indexes devicetrust

Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen P
devicetrust	Edit Delete Disable	Events	search	1 MB	500 GB	0			$SPLUNK_DB/devicetrust/db	N/A

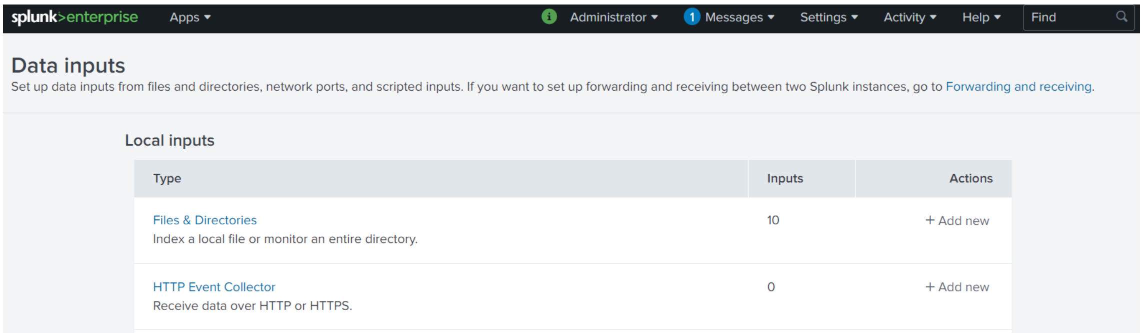
Step 1.2: Creating the Splunk Index by installing the app

To create the index from the app, please refer to Step 3.2: Installing the Splunk app within this guide. The app `dt_index` does not contain any elements besides the index definition for the index `devicetrust`.

Step 2: Creating the Splunk Data Inputs

Data is sent to the Splunk server by using REST API calls. Splunk needs to be configured to accept http-based inputs. An authentication token is generated, that will be added to the deviceTRUST Console configuration later.

- Open your splunk management GUI and navigate to `Settings\Data Inputs`.



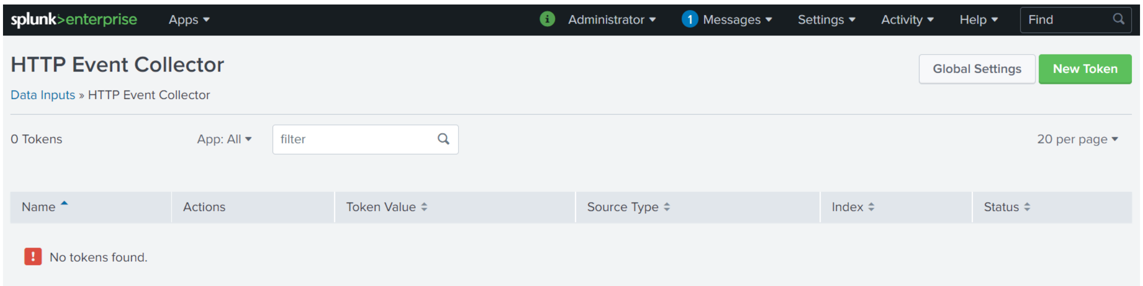
Data inputs

Set up data inputs from files and directories, network ports, and scripted inputs. If you want to set up forwarding and receiving between two Splunk instances, go to Forwarding and receiving.

Local inputs

Type	Inputs	Actions
Files & Directories Index a local file or monitor an entire directory.	10	+ Add new
HTTP Event Collector Receive data over HTTP or HTTPS.	0	+ Add new

- Select `HTTP Event Collector`
- Click 'New Token



HTTP Event Collector

Data Inputs » HTTP Event Collector

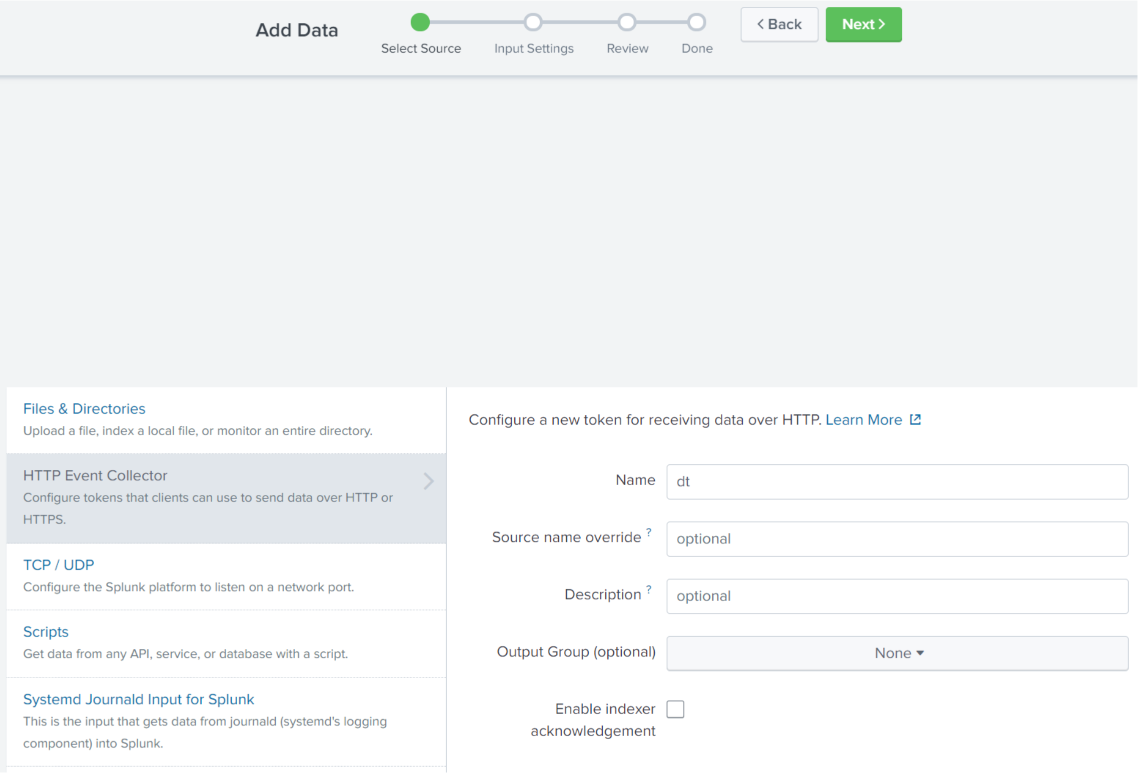
0 Tokens

Name	Actions	Token Value	Source Type	Index	Status

No tokens found.

- Set a `name` of your choice.

- All other settings are optional.



Files & Directories
Upload a file, index a local file, or monitor an entire directory.

HTTP Event Collector
Configure tokens that clients can use to send data over HTTP or HTTPS.

TCP / UDP
Configure the Splunk platform to listen on a network port.

Scripts
Get data from any API, service, or database with a script.

Systemd Journald Input for Splunk
This is the input that gets data from journald (systemd's logging component) into Splunk.

Configure a new token for receiving data over HTTP. Learn More

Name: dt

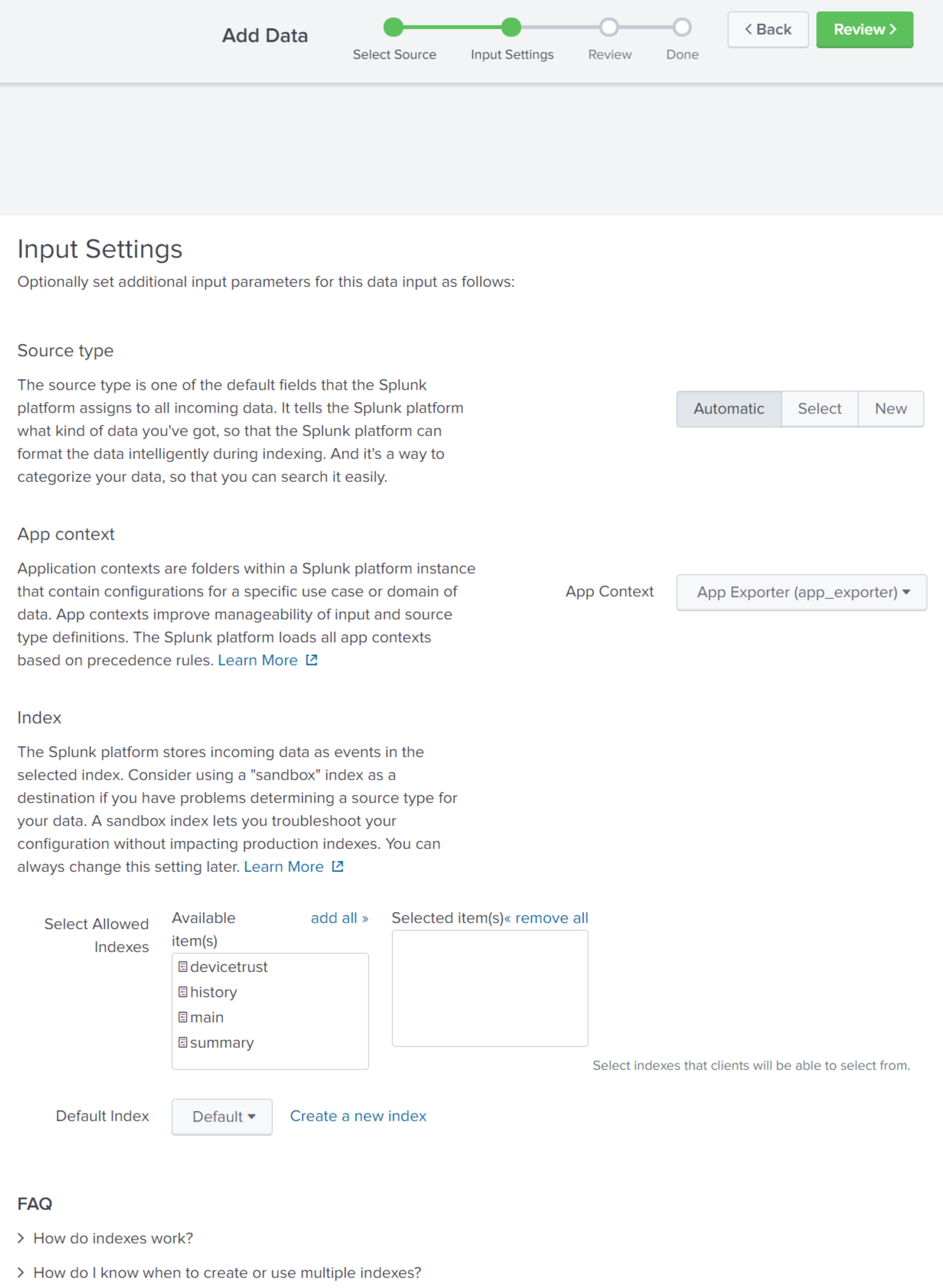
Source name override: optional

Description: optional

Output Group (optional): None

Enable indexer acknowledgement □

`Input settings` does, for example, allow to restrict access for this data input to certain indexes. Any of these settings may be relevant for your environment. The function of the deviceTRUST reports will not be affected by setting them.



Input Settings

Optionally set additional input parameters for this data input as follows:

Source type

The source type is one of the default fields that the Splunk platform assigns to all incoming data. It tells the Splunk platform what kind of data you've got, so that the Splunk platform can format the data intelligently during indexing. And it's a way to categorize your data, so that you can search it easily.

App context

Application contexts are folders within a Splunk platform instance that contain configurations for a specific use case or domain of data. App contexts improve manageability of input and source type definitions. The Splunk platform loads all app contexts based on precedence rules. Learn More

App Context: App Exporter (app_exporter)

Index

The Splunk platform stores incoming data as events in the selected index. Consider using a "sandbox" index as a destination if you have problems determining a source type for your data. A sandbox index lets you troubleshoot your configuration without impacting production indexes. You can always change this setting later. Learn More

Select Allowed Indexes

Available item(s)

- devicetrust
- history
- main
- summary

Selected item(s)

Select indexes that clients will be able to select from.

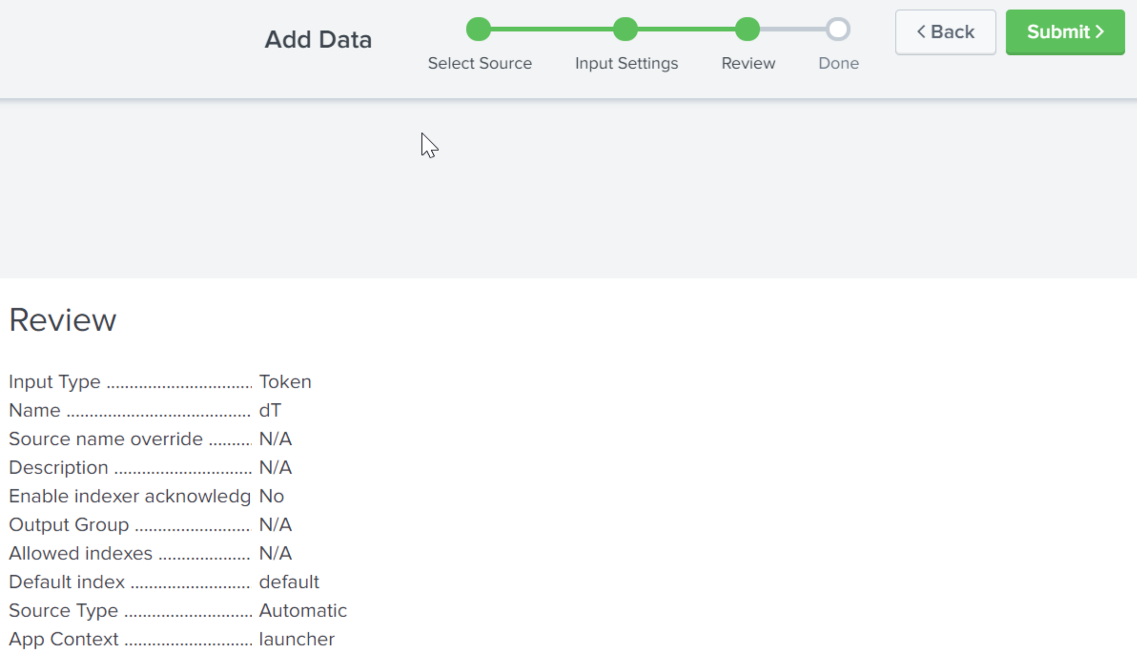
Default Index: Default

FAQ

> How do indexes work?

> How do I know when to create or use multiple indexes?

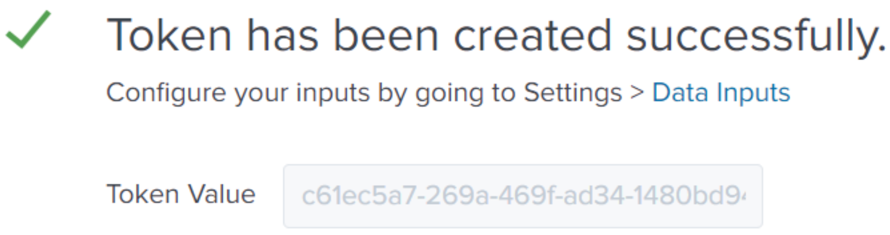
- Review the Settings.
- Click `Submit`.



Review

Input Type .... Token
Name .... dT
Source name override .... N/A
Description .... N/A
Enable indexer acknowledg No
Output Group .... N/A
Allowed indexes .... N/A
Default index .... default
Source Type .... Automatic
App Context .... launcher

- The created token is shown.

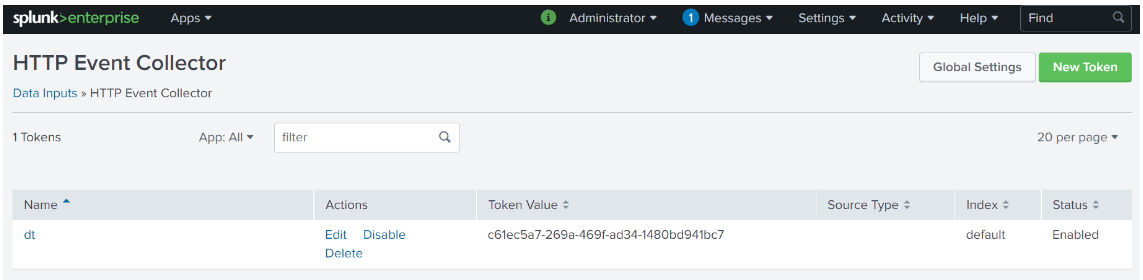


Token has been created successfully.

Configure your inputs by going to Settings > Data Inputs

Token Value c61ec5a7-269a-469f-ad34-1480bd9

- The created token is also be displayed in the token overview page.



HTTP Event Collector

Data Inputs » HTTP Event Collector

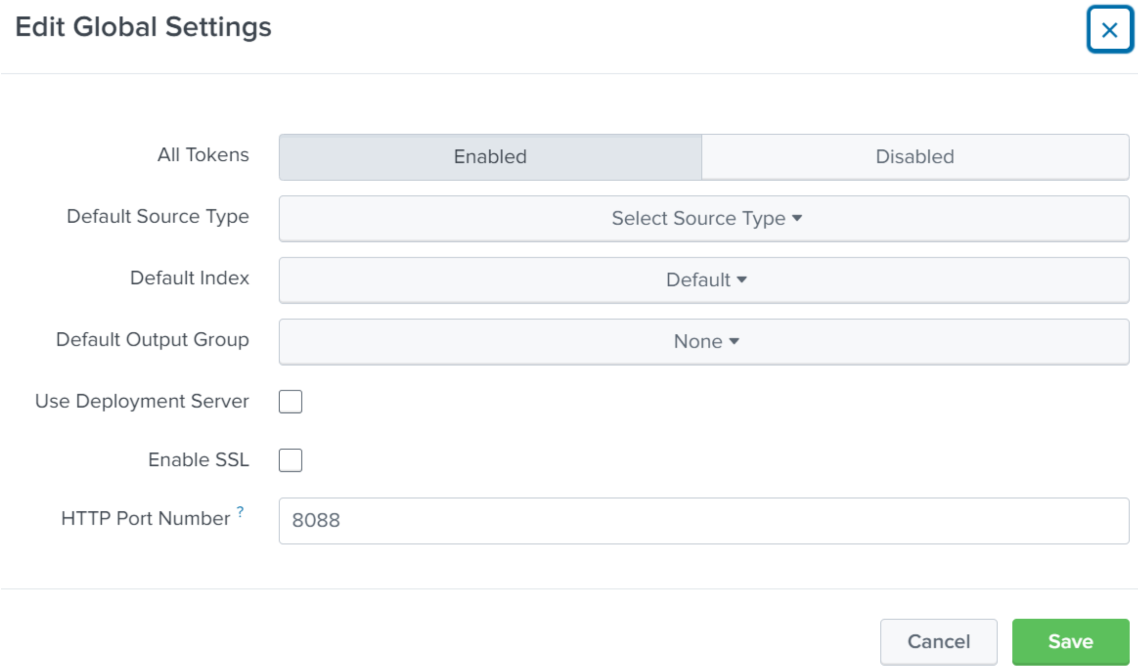
1 Tokens

Name	Actions	Token Value	Source Type	Index	Status
dt	Edit Disable Delete	c61ec5a7-269a-469f-ad34-1480bd941bc7		default	Enabled

Additionally, the http input method has to be configured. In the most basic configuration, we make sure SSL is not active and all tokens are enabled.

These settings may differ in your environment. The function of the deviceTRUST reports will not be affected by setting them accordingly.

- Open your splunk management GUI and navigate to `Settings\Data Inputs\HTTP Data Collector`.
- Click `Global Settings`
- Set `All Tokens` to Enabled.
- Set `Enable SSL` to Off.



Edit Global Settings

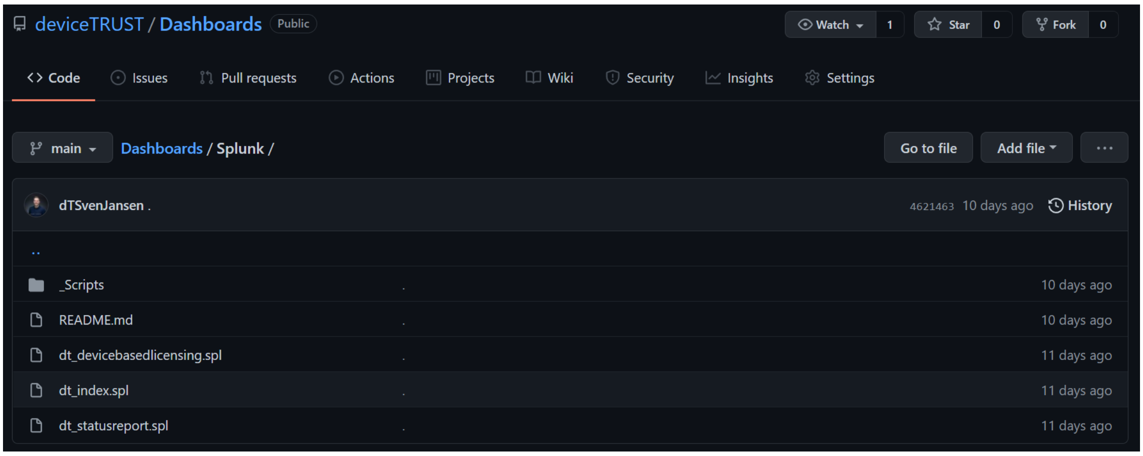
All Tokens	Enabled / Disabled
Default Source Type	Select Source Type
Default Index	Default
Default Output Group	None
Use Deployment Server	□
Enable SSL	□
HTTP Port Number ?	8088

Step 3: Importing the Splunk app

Step 3.1: Downloading the Splunk app

The deviceTRUST reports are delivered as Splunk apps. The apps is available from the deviceTRUST GitHub repository.

- Navigate to the repository and find the Splunk apps.

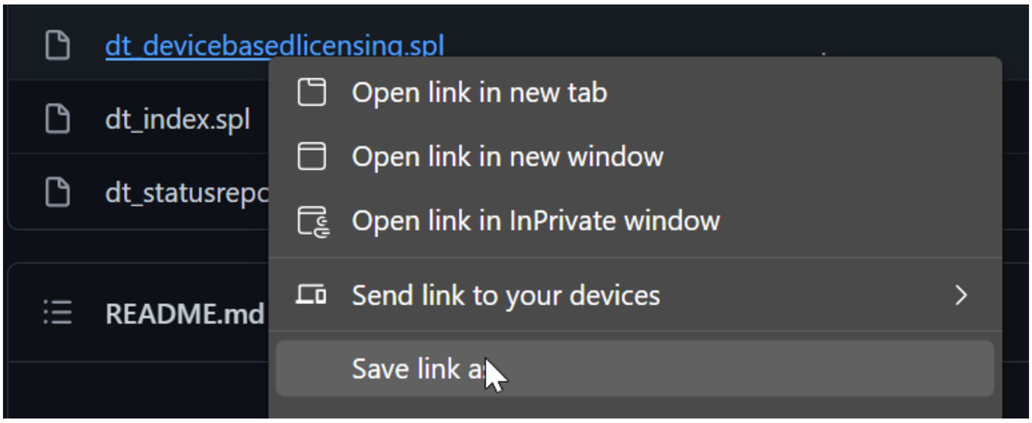


dTSvenJansen . 4621463 10 days ago

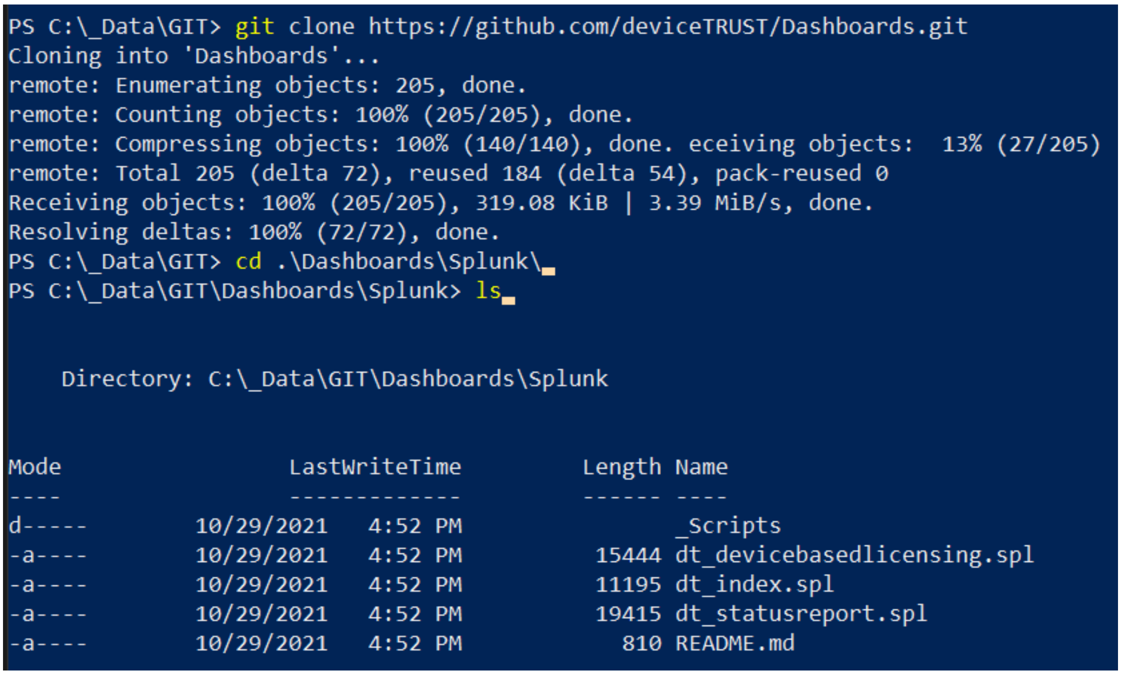
..

_Scripts	.	10 days ago
README.md	.	10 days ago
dt_devicebasedlicensing.spl	.	11 days ago
dt_index.spl	.	11 days ago
dt_statusreport.spl	.	11 days ago

- The apps are provided as “spl”files which should be downloaded.



- Alternatively the “spl”files can be synchronized via the GIT command line.

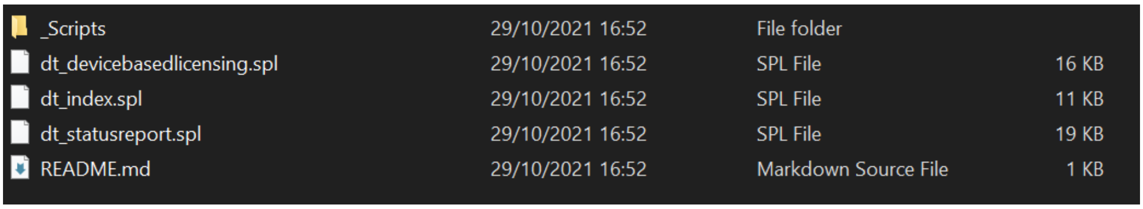


```
PS C:\_Data\GIT> git clone https://github.com/deviceTRUST/Dashboards.git
Cloning into 'Dashboards'...
remote: Enumerating objects: 205, done.
remote: Counting objects: 100% (205/205), done.
remote: Compressing objects: 100% (140/140), done. eceiving objects:  13% (27/205)
remote: Total 205 (delta 72), reused 184 (delta 54), pack-reused 0
Receiving objects: 100% (205/205), 319.08 KiB | 3.39 MiB/s, done.
Resolving deltas: 100% (72/72), done.
PS C:\_Data\GIT> cd .\Dashboards\Splunk\
PS C:\_Data\GIT\Dashboards\Splunk> ls


    Directory: C:\_Data\GIT\Dashboards\Splunk


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        10/29/2021   4:52 PM                _Scripts
-a----        10/29/2021   4:52 PM          15444 dt_devicebasedlicensing.spl
-a----        10/29/2021   4:52 PM          11195 dt_index.spl
-a----        10/29/2021   4:52 PM          19415 dt_statusreport.spl
-a----        10/29/2021   4:52 PM            810 README.md
```

- With the apps available locally, they can be imported into Splunk.

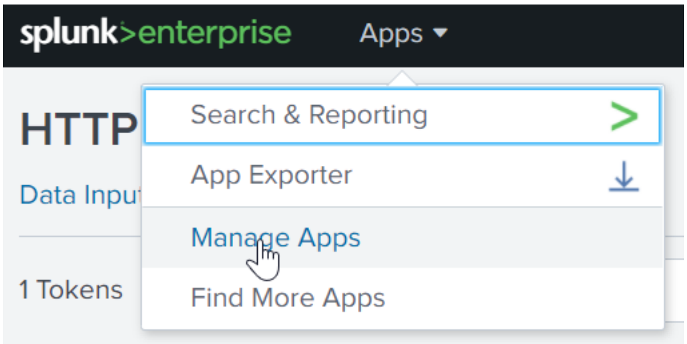


_Scripts	29/10/2021 16:52	File folder	
dt_devicebasedlicensing.spl	29/10/2021 16:52	SPL File	16 KB
dt_index.spl	29/10/2021 16:52	SPL File	11 KB
dt_statusreport.spl	29/10/2021 16:52	SPL File	19 KB
README.md	29/10/2021 16:52	Markdown Source File	1 KB

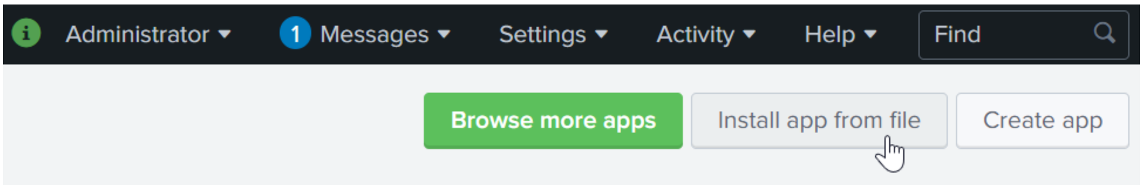
Step 3.2: Installing the Splunk app

All three deviceTRUST apps are installed the same way. Thus, the app installation is described by using one example.

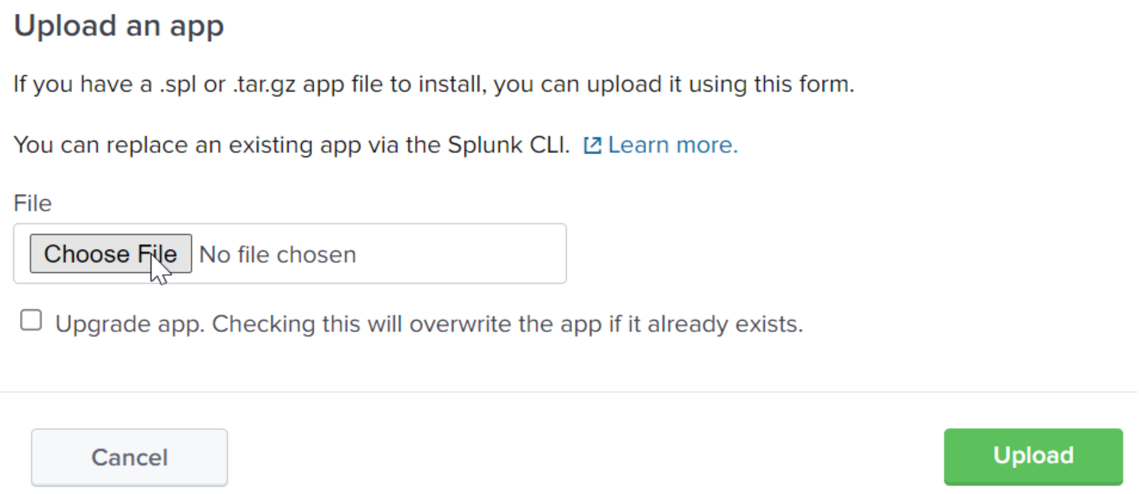
- Open your Splunk management console.
- Click `Manage Apps` in the apps menu.



- Click `Install app from file`.



- Click `Choose File`.



Upload an app

If you have a .spl or .tar.gz app file to install, you can upload it using this form.

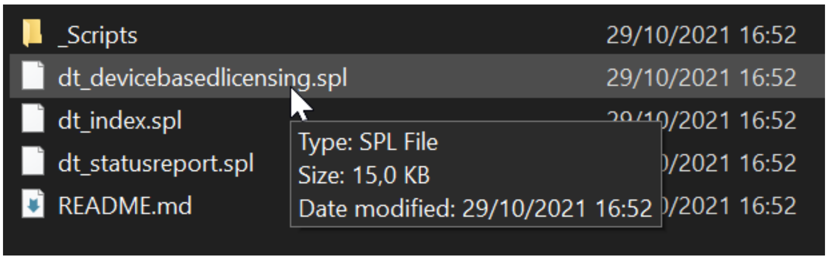
You can replace an existing app via the Splunk CLI. Learn more.

File

Choose File No file chosen

□ Upgrade app. Checking this will overwrite the app if it already exists.

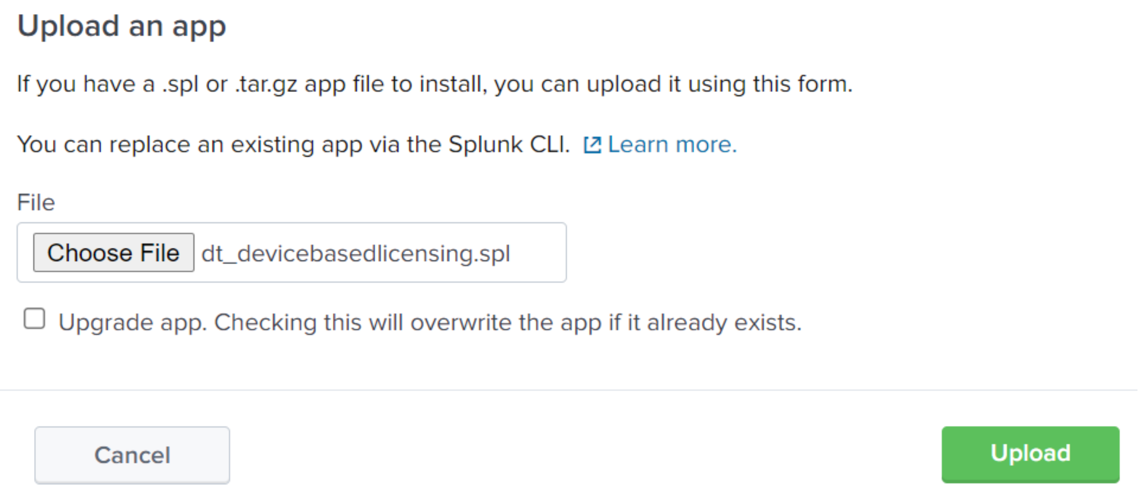
- Select your app file.



_Scripts	29/10/2021 16:52
dt_devicebasedlicensing.spl	29/10/2021 16:52
dt_index.spl	29/10/2021 16:52
dt_statusreport.spl	)/2021 16:52
README.md	)/2021 16:52

Type: SPL File
Size: 15,0 KB
Date modified: 29/10/2021 16:52

- Confirming your selection.
- Optionally select to upgrade apps, if applicable.
- Click `Upload`.



Upload an app

If you have a .spl or .tar.gz app file to install, you can upload it using this form.

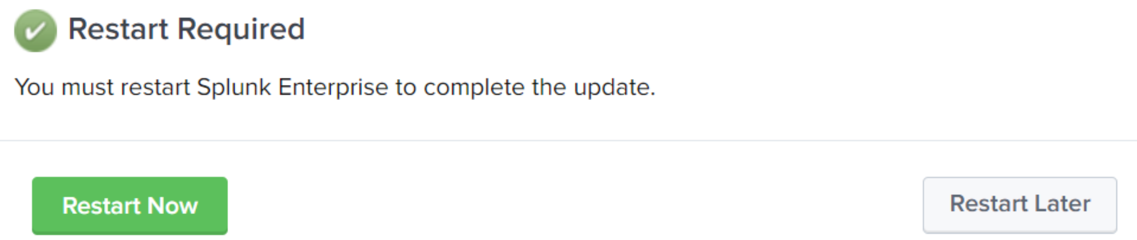
You can replace an existing app via the Splunk CLI. Learn more.

File

Choose File dt_devicebasedlicensing.spl

□ Upgrade app. Checking this will overwrite the app if it already exists.

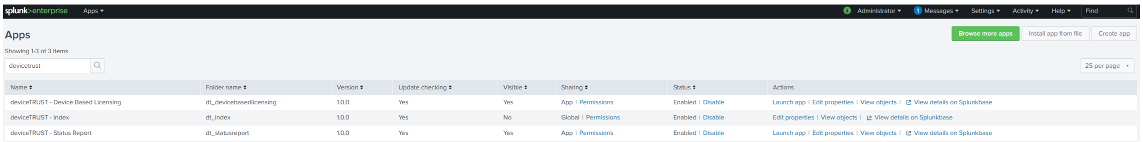
- Splunk will ask to restart the service. This is only required after importing the last app.
- Choose `Restart Now` or `Restart Later` accordingly.



Restart Required

You must restart Splunk Enterprise to complete the update.

- The menu Apps\Manage Apps displays all apps that are installed in your Splunk environment.



Apps

Showing 1-3 of 3 items

devicetrust

Name	Folder name	Version	Update checking	Visible	Sharing	Status	Actions
deviceTRUST - Device Based Licensing	dt_devicebasedlicensing	1.0.0	Yes	Yes	App | Permissions	Enabled | Disable	Launch app | Edit properties | View objects | View details on Splunkbase
deviceTRUST - Index	dt_index	1.0.0	Yes	No	Global | Permissions	Enabled | Disable	Edit properties | View objects | View details on Splunkbase
deviceTRUST - Status Report	dt_statusreport	1.0.0	Yes	Yes	App | Permissions	Enabled | Disable	Launch app | Edit properties | View objects | View details on Splunkbase

Step 4: Configuring deviceTRUST

After the the index has been created, the data input object added and all apps have been imported, Splunk is ready to accept, store and compute data for the deviceTRUST reports.

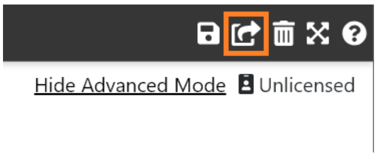
As both reports Device-Based Licensing and Status Report differ in their details, both are described here separately.

Step 4.1: The Device-Based Licensing Report

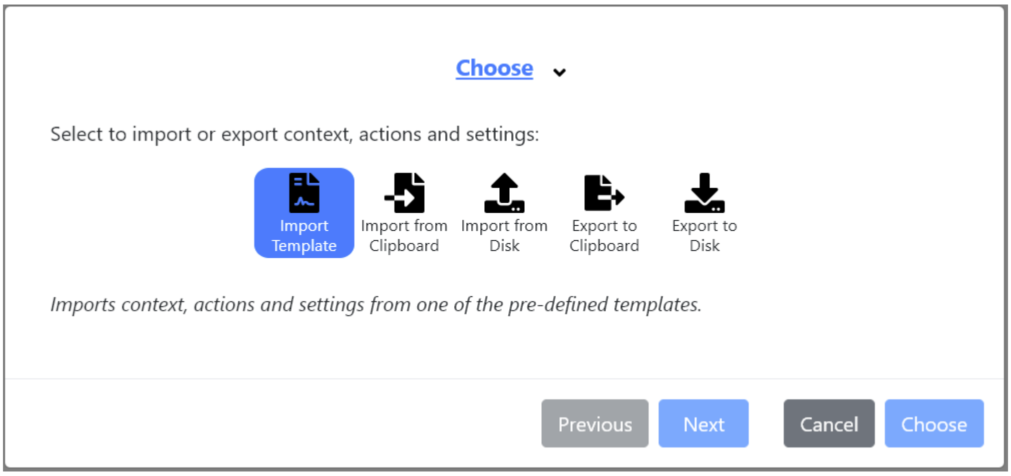
This step describes the configuration to be added to the deviceTRUST Console to send Device-Based Licensing data to Splunk.

The integrated templates contain all elements that are required to fully configure the agent for the Device-Based Licensing of five example applications. These five example applications can be easily edited for your own applications, or cloned to represent new applications. The elements are contexts, actions, messages and settings.

- Open the deviceTRUST Console.
- Click Sharing in the top right menu. You may need to click Show Advanced View if this button is not visible.



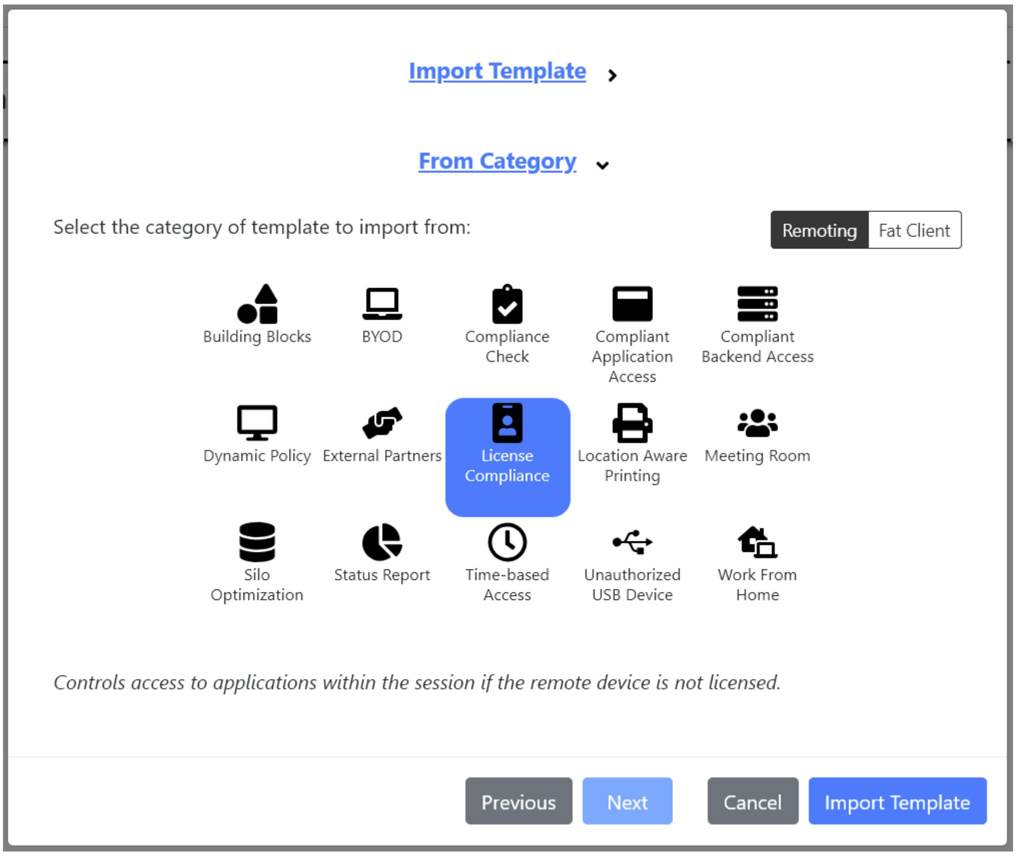
- Select Import Template



Select to import or export context, actions and settings:

Imports context, actions and settings from one of the pre-defined templates.

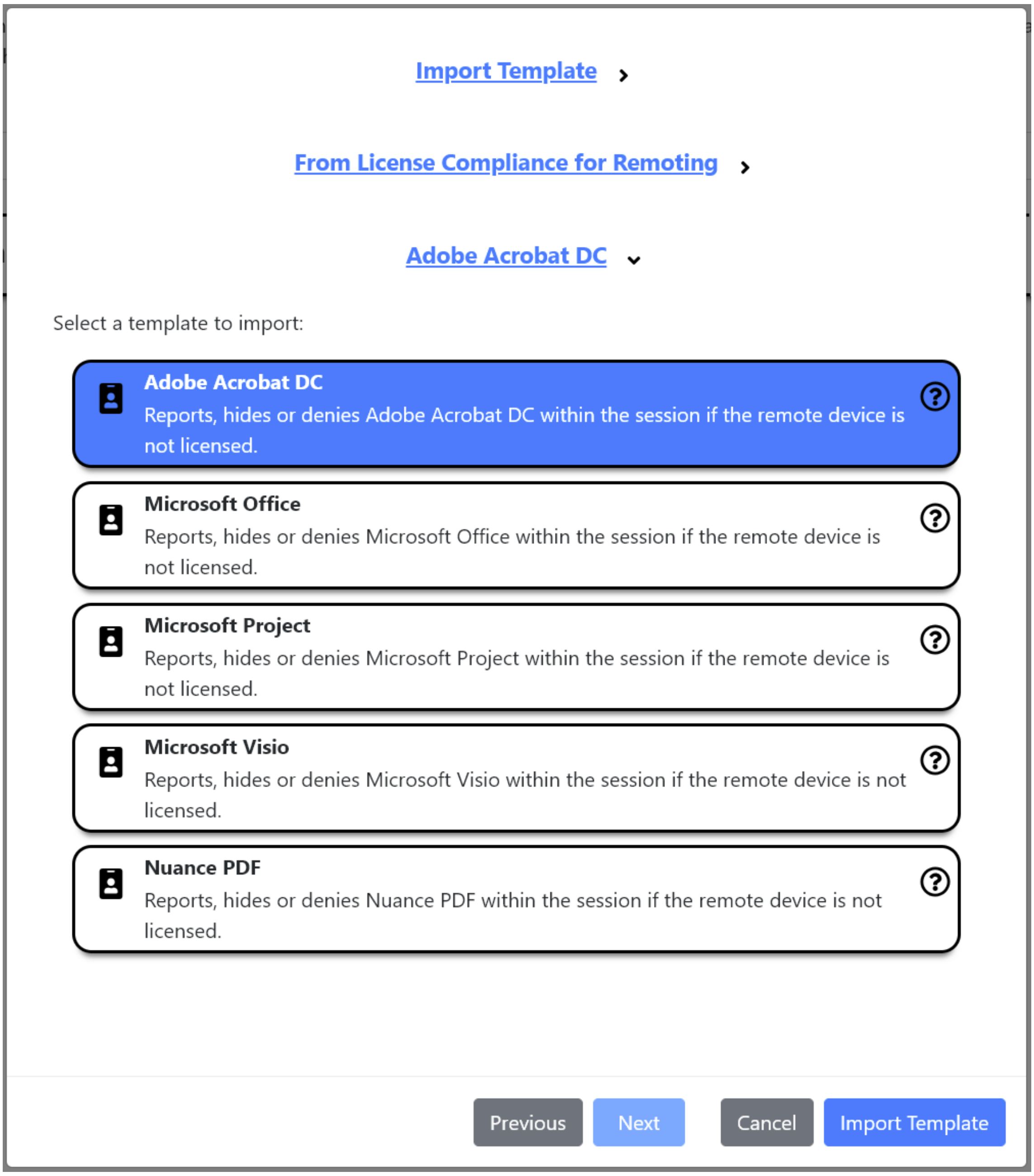
- The report `Device-Based Licensing` can be found within the template category `License Compliance` when `Remoting` is selected.



Select the category of template to import from:

Controls access to applications within the session if the remote device is not licensed.

- The category contains templates for different example applications. We use `Adobe Acrobat DC` as an example here.



Select a template to import:

Adobe Acrobat DC
Reports, hides or denies Adobe Acrobat DC within the session if the remote device is not licensed.

Microsoft Office
Reports, hides or denies Microsoft Office within the session if the remote device is not licensed.

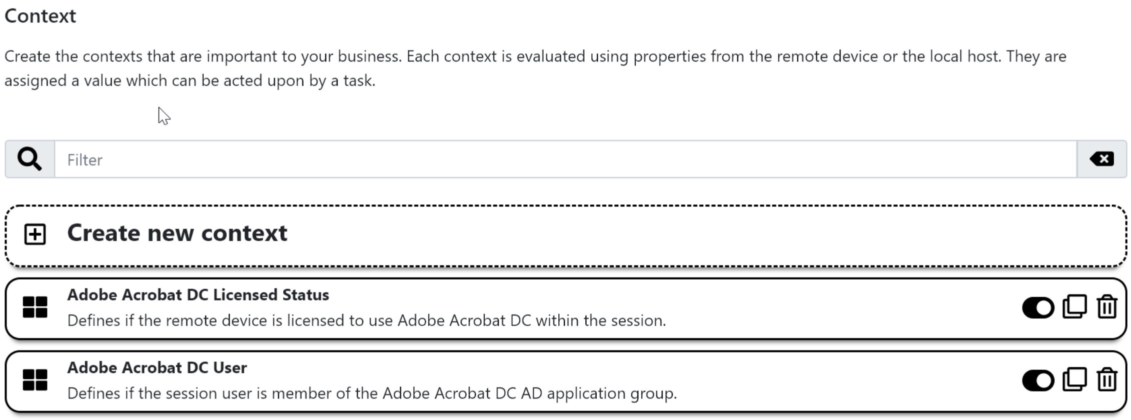
Microsoft Project
Reports, hides or denies Microsoft Project within the session if the remote device is not licensed.

Microsoft Visio
Reports, hides or denies Microsoft Visio within the session if the remote device is not licensed.

Nuance PDF
Reports, hides or denies Nuance PDF within the session if the remote device is not licensed.

Two contexts are included within the template:

- `Adobe Acrobat DC Licensed Status` to evaluate the device license status.
- `Adobe Acrobat DC User` to define if the accessing user is licensed to use the software.



Context

Create the contexts that are important to your business. Each context is evaluated using properties from the remote device or the local host. They are assigned a value which can be acted upon by a task.

Filter

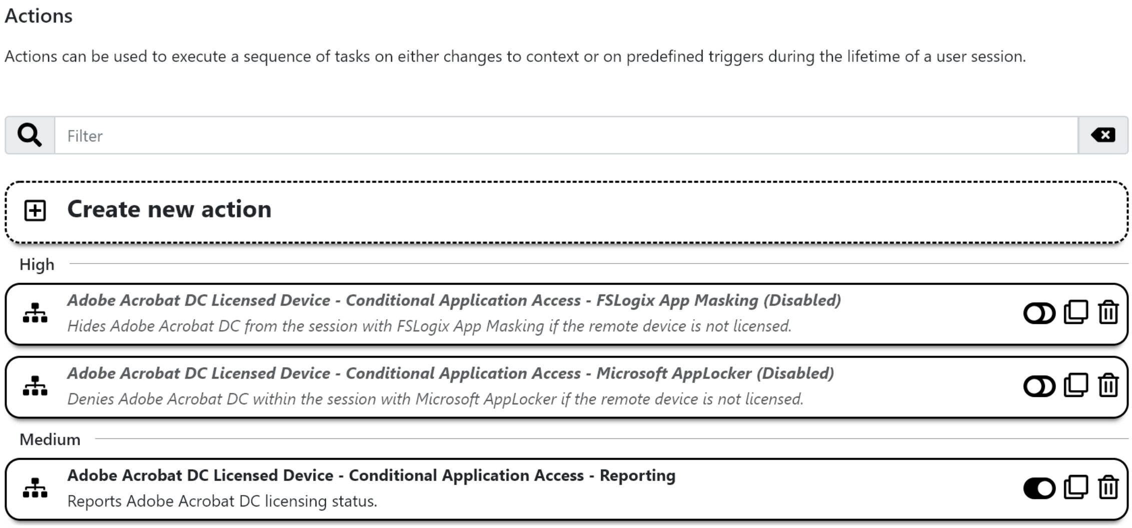
Create new context

Adobe Acrobat DC Licensed Status
Defines if the remote device is licensed to use Adobe Acrobat DC within the session.

Adobe Acrobat DC User
Defines if the session user is member of the Adobe Acrobat DC AD application group.

Three actions are included within the template:

- `Adobe Acrobat DC Licensed Device - Conditional Application Access - FSLogix App Masking` is used for application control via `FSLogix App Masking` and can be ignored or deleted for the reporting use case.
- `Adobe Acrobat DC Licensed Device - Conditional Application Access - Microsoft AppLocker` is used for application control via `Microsoft AppLocker` and can be ignored or deleted for the reporting use case.
- `Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting` the only action required for reporting.



Actions

Actions can be used to execute a sequence of tasks on either changes to context or on predefined triggers during the lifetime of a user session.

Filter

Create new action

High

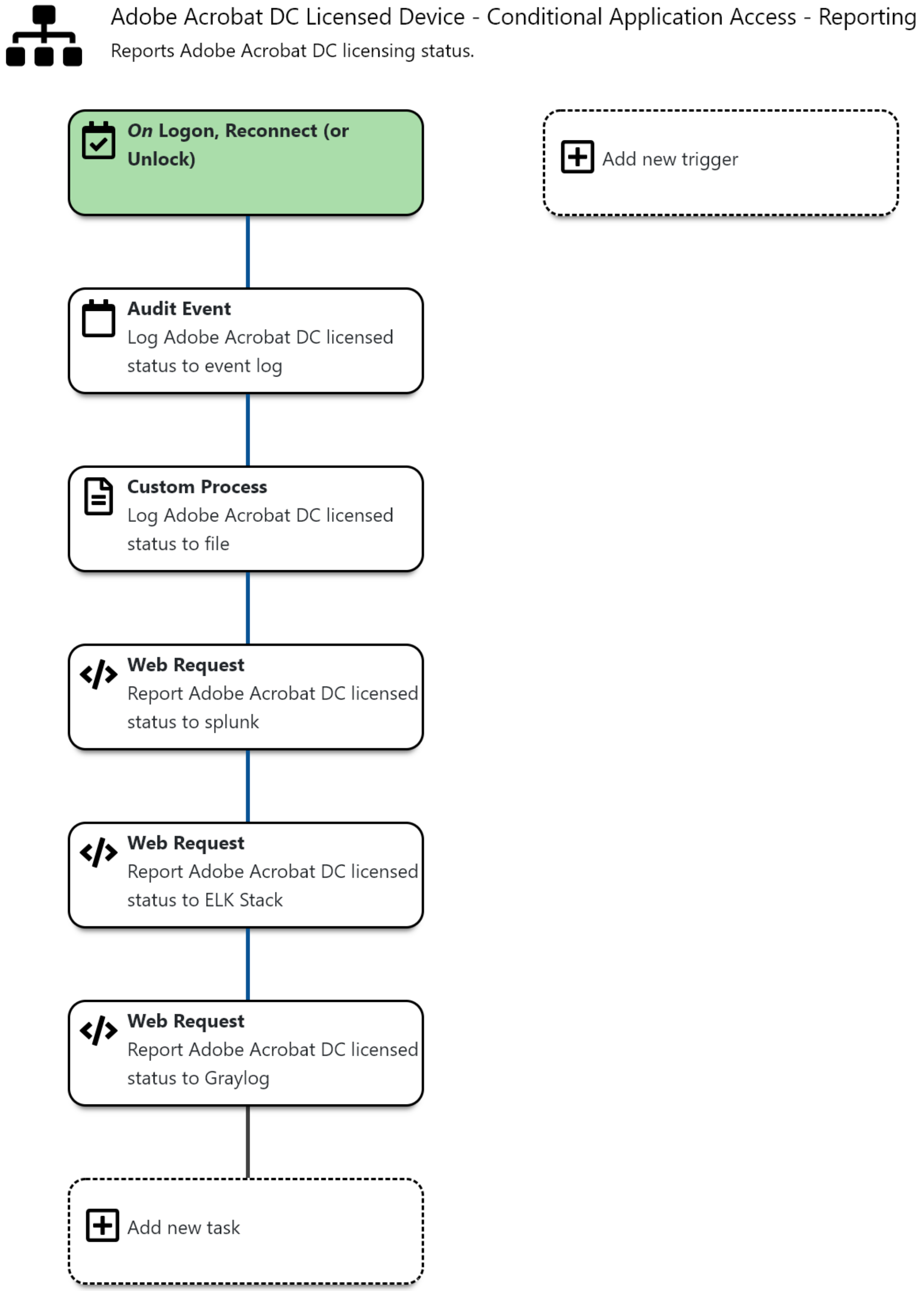
Adobe Acrobat DC Licensed Device - Conditional Application Access - FSLogix App Masking (Disabled)
Hides Adobe Acrobat DC from the session with FSLogix App Masking if the remote device is not licensed.

Adobe Acrobat DC Licensed Device - Conditional Application Access - Microsoft AppLocker (Disabled)
Denies Adobe Acrobat DC within the session with Microsoft AppLocker if the remote device is not licensed.

Medium

Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting
Reports Adobe Acrobat DC licensing status.

- The action contains uses the `Web Request` task to send data to Splunk. The `Audit Event`, `Custom Process` and also the `Web Request` tasks for ELK Stack and Graylog can be deleted, as we are configuring Splunk here.



Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting

Reports Adobe Acrobat DC licensing status.

***On* Logon, Reconnect (or Unlock)**

⊞ Add new trigger

Audit Event
Log Adobe Acrobat DC licensed status to event log

Custom Process
Log Adobe Acrobat DC licensed status to file

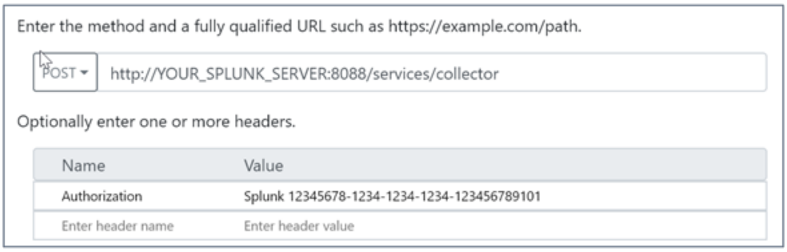
Web Request
Report Adobe Acrobat DC licensed status to splunk

Web Request
Report Adobe Acrobat DC licensed status to ELK Stack

Web Request
Report Adobe Acrobat DC licensed status to Graylog

⊞ Add new task

- The Splunk `Web Request` task needs to be edited to suit your environment. If you do not require SSL transport, you'll only have to configure your `server's fqdn` and the `authorization token` (http data input). Make sure to leave the keyword `splunk` and only add/change your authorization token's GUID.



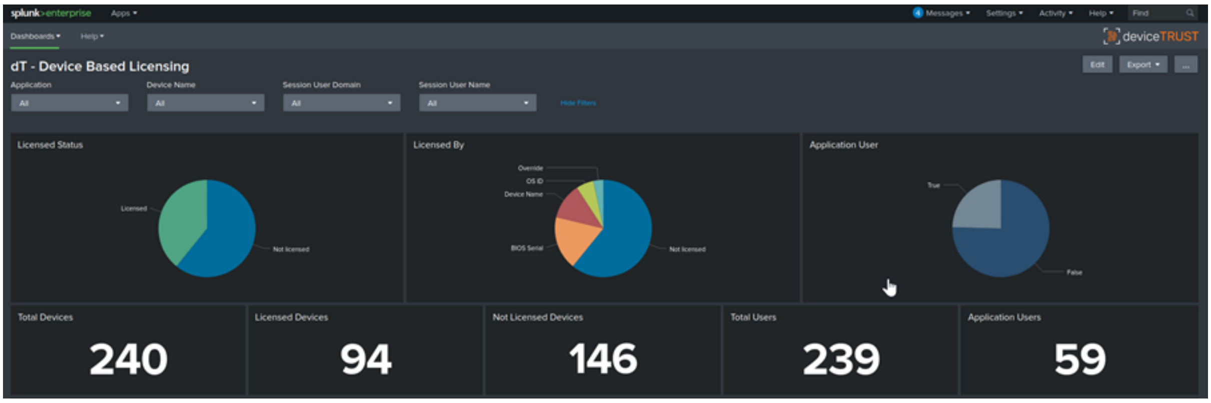
Enter the method and a fully qualified URL such as https://example.com/path.

POST ▾ http://YOUR_SPLUNK_SERVER:8088/services/collector

Optionally enter one or more headers.

Name	Value
Authorization	Splunk 12345678-1234-1234-1234-123456789101
Enter header name	Enter header value

The use case has successfully been configured. deviceTRUST will now send `Device-Based Licensing` data on every access to the Splunk server. The data will be presented using dashboards.

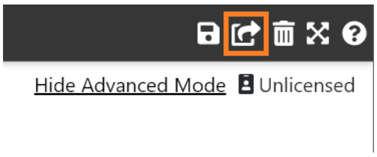


Step 4.2: The Status Report

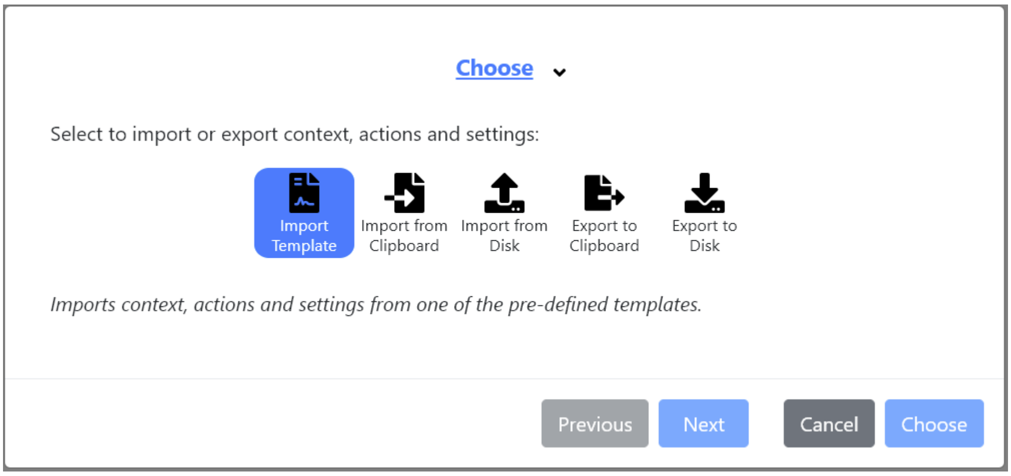
This step describes the configuration, that is to be added to the deviceTRUST console to send `Status Report` data to splunk.

Our integrated templates contain all elements that are required to fully configure the agent for the Device-Based Licensing of five example applications. The elements are contexts, actions, messages and settings.

- Open the deviceTRUST console and click `Sharing` in the top right menu.



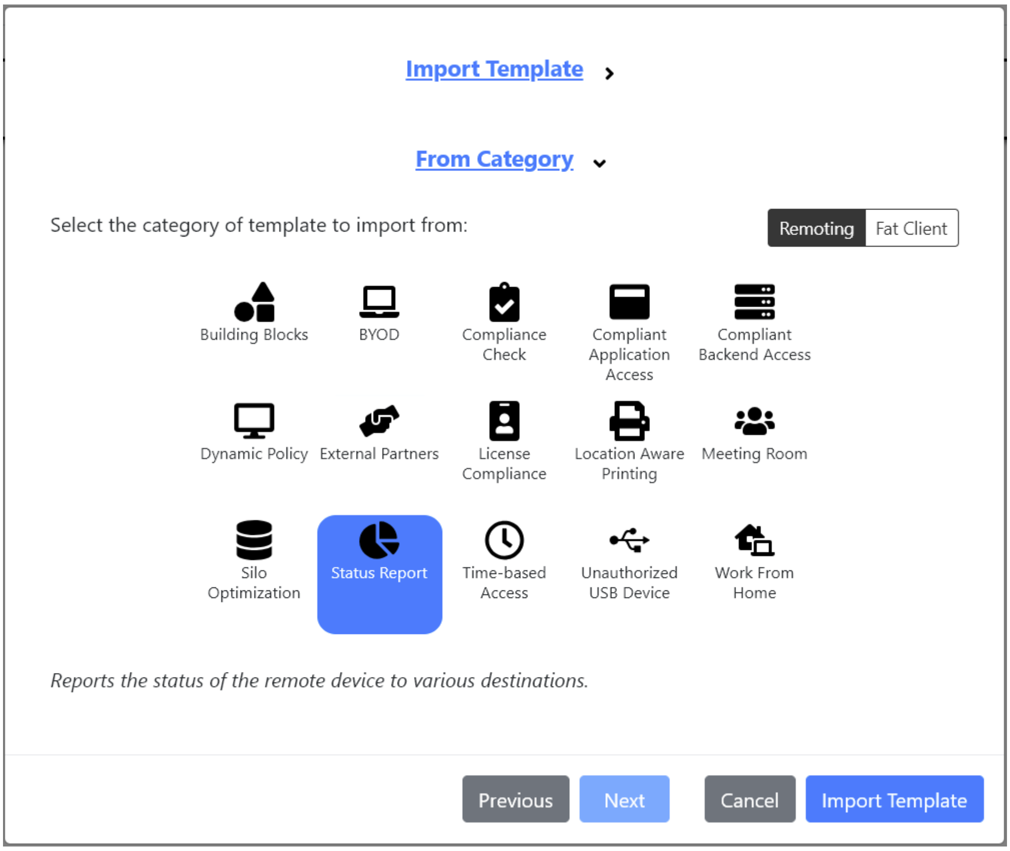
- Select `Import Template`



Select to import or export context, actions and settings:

Imports context, actions and settings from one of the pre-defined templates.

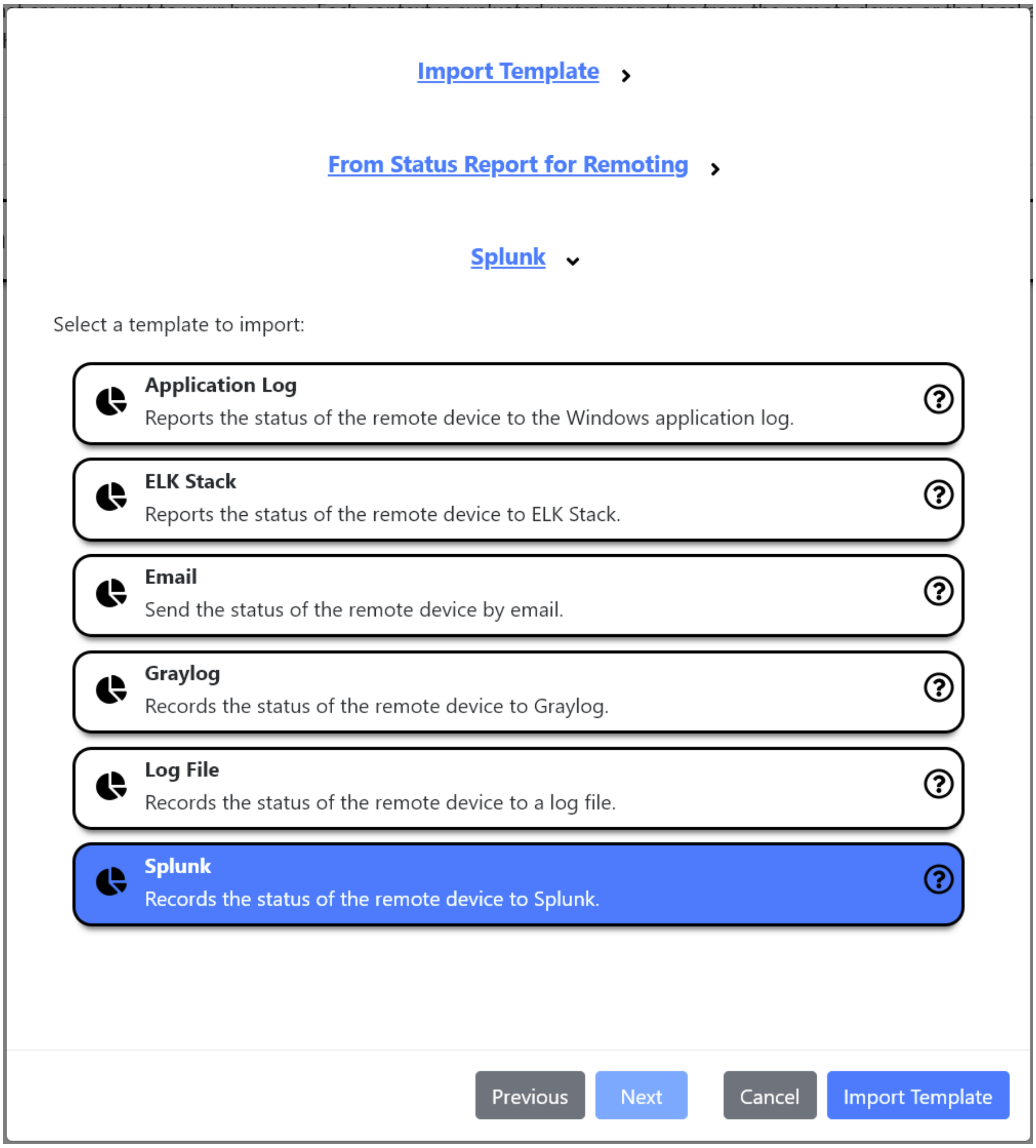
- The `Status Report` template can be found within the template category `Status Report` when `Remoting` is selected.



Select the category of template to import from:

Reports the status of the remote device to various destinations.

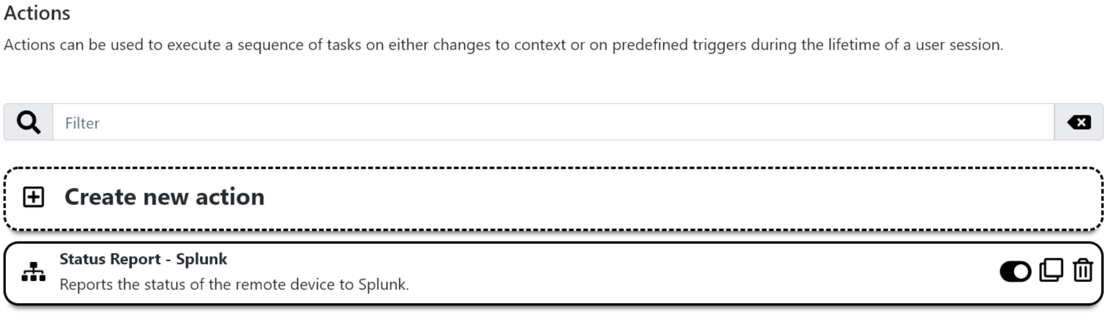
- This category contains templates for several ways of storing data. Choose `Splunk`.



Select a template to import:

- **Application Log** Reports the status of the remote device to the Windows application log.
- **ELK Stack** Reports the status of the remote device to ELK Stack.
- **Email** Send the status of the remote device by email.
- **Graylog** Records the status of the remote device to Graylog.
- **Log File** Records the status of the remote device to a log file.
- **Splunk** Records the status of the remote device to Splunk.

- The imported template consists of 50 contexts and one action.
- The Action `Status Report – Splunk` collects all relevant data and sends them over to Splunk.



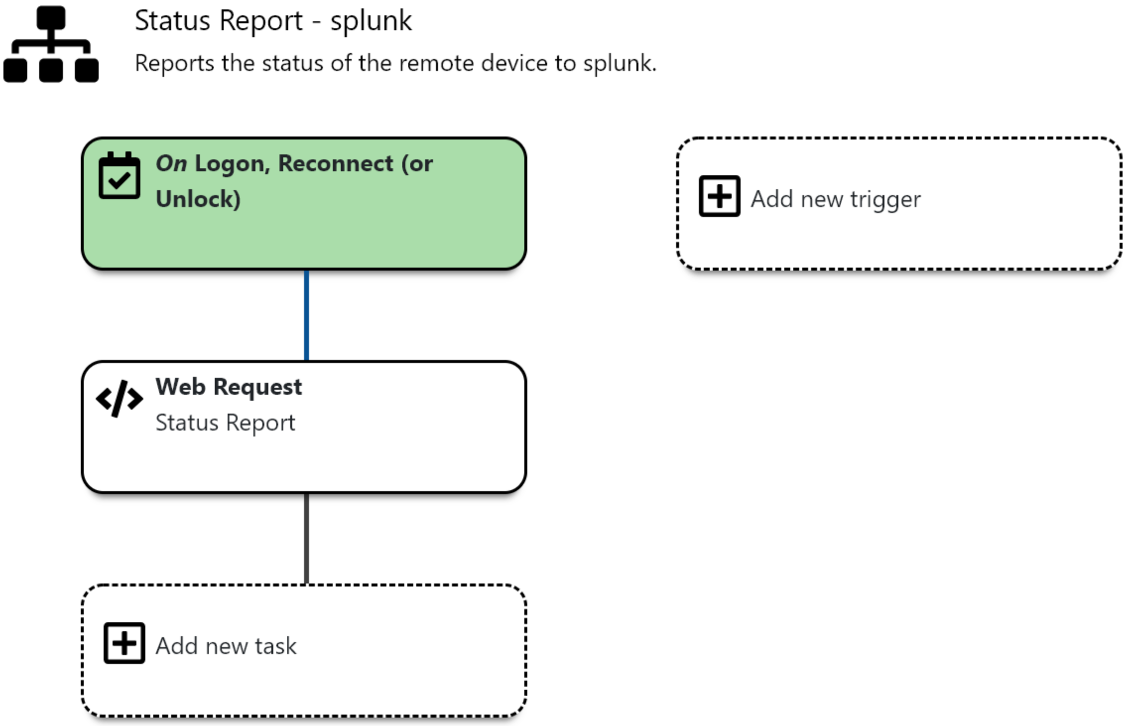
Actions

Actions can be used to execute a sequence of tasks on either changes to context or on predefined triggers during the lifetime of a user session.

Status Report - Splunk

Reports the status of the remote device to Splunk.

- Sending data to Splunk is configured by using a `Web Request` task.



Status Report - splunk

Reports the status of the remote device to splunk.

The Web Request task needs to be edited to suit your environment. If you do not require SSL transport, you'll only have to configure your `server's fqdn` and the `authorization token` (http data input). Make sure to leave the keyword `splunk` and only change your authorization token's GUID.

When executing Web Request task ›

Apply settings ⌄

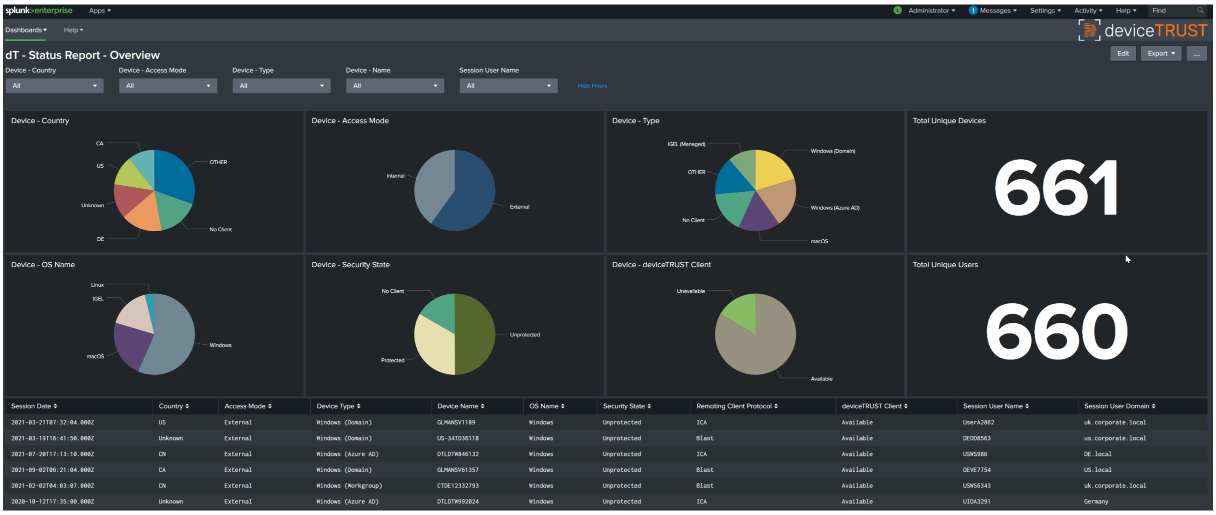
Enter the method and a fully qualified URL such as https://example.com/path.

POST ▾ http://YOUR_SPLUNK_SERVER:8088/services/collector

Optionally enter one or more headers.

Name	Value
Authorization	Splunk YOUR_AUTH_TOKEN
Enter header name	Enter header value

The use case has successfully been configured. deviceTRUST will now send `Status Report` data on every access to the remoting platform to the splunk server. There, the data will be presented using dashboards.



dT - Status Report - Overview

Device - Country: All
Device - Access Mode: All
Device - Type: All
Device - Name: All
Session User Name: All

Device - Country

Device - Access Mode

Device - Type

Total Unique Devices: 661

Device - OS Name

Device - Security State

Device - deviceTRUST Client

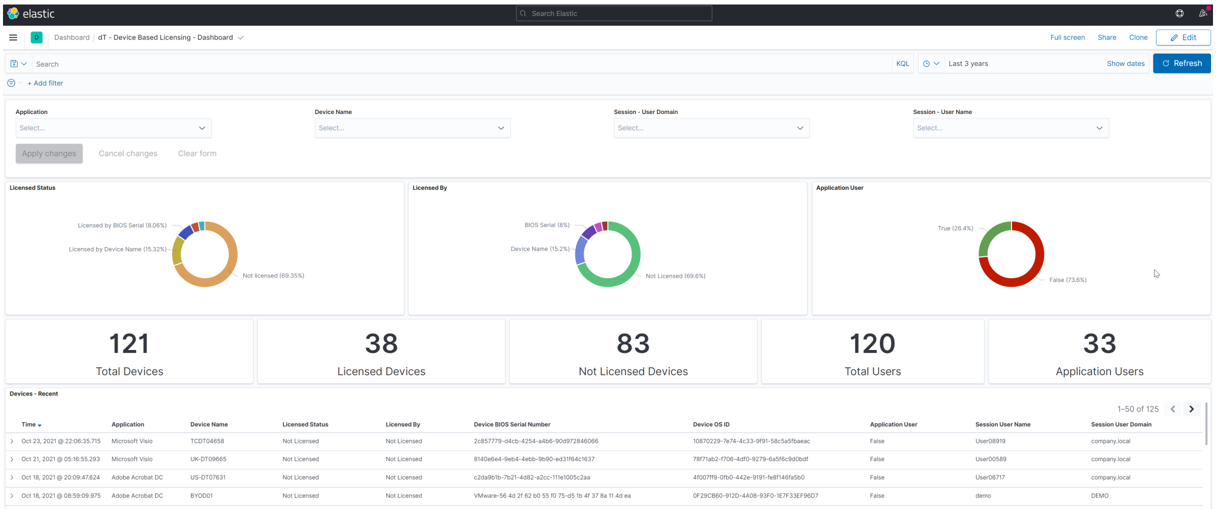
Total Unique Users: 660

Session Date ⇕	Country ⇕	Access Mode ⇕	Device Type ⇕	Device Name ⇕	OS Name ⇕	Security State ⇕	Remoting Client Protocol ⇕	deviceTRUST Client ⇕	Session User Name ⇕	Session User Domain ⇕
2021-03-21T07:32:04.000Z	US	External	Windows (Domain)	GLMANSV1189	Windows	Unprotected	ICA	Available	UserA2862	uk.corporate.local
2021-03-19T16:41:50.000Z	Unknown	External	Windows (Domain)	US-34TD36118	Windows	Unprotected	Blast	Available	DEDD8563	us.corporate.local
2021-07-20T17:13:10.000Z	CN	External	Windows (Azure AD)	DTLDTW846132	Windows	Unprotected	ICA	Available	USWS986	DE.local
2021-09-02T06:21:04.000Z	CA	External	Windows (Domain)	GLMANSV61357	Windows	Unprotected	Blast	Available	DEVE7754	US.local
2021-02-02T04:03:07.000Z	CN	External	Windows (Workgroup)	CTDE12332793	Windows	Unprotected	Blast	Available	USWS6343	uk.corporate.local
2020-10-12T17:35:00.000Z	Unknown	External	Windows (Azure AD)	DTLDTW992024	Windows	Unprotected	ICA	Available	UIDA3291	Germany

ELK Stack Dashboards

deviceTRUST® includes components for an ELK Stack to easily create a dashboard to monitor the contextual status of your remoting and DaaS environment.

The License Compliance Templates can be used with your ELK Stack to monitor or enforce Device-Based Licensing requirements for one or more applications:



121 Total Devices

38 Licensed Devices

83 Not Licensed Devices

120 Total Users

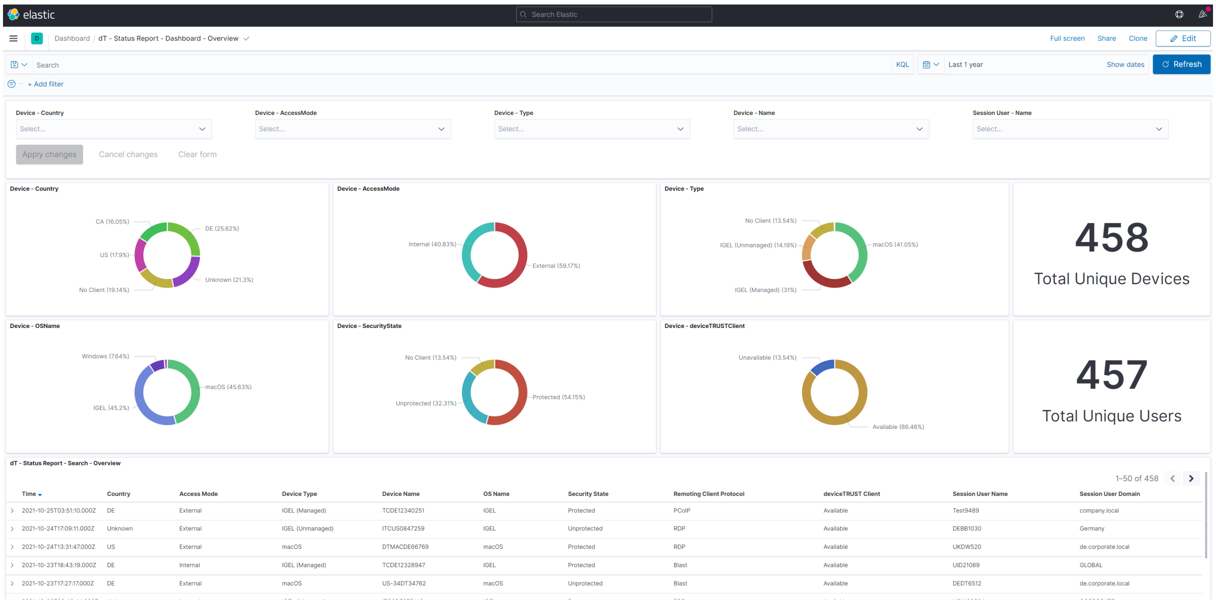
33 Application Users

Devices - Recent

1–50 of 125

Time	Application	Device Name	Licensed Status	Licensed By	Device BIOS Serial Number	Device OS ID	Application User	Session User Name	Session User Domain
Oct 23, 2021 @ 22:06:35.715	Microsoft Visio	TCDT04658	Not Licensed	Not Licensed	2c857779-d4cb-4254-a4b6-90d972846066	10870229-7e74-4c33-9f91-58c5a5fbaeac	False	User08919	company.local
Oct 21, 2021 @ 05:16:55.293	Microsoft Visio	UK-DT09665	Not Licensed	Not Licensed	8140e6e4-9eb4-4ebb-9b90-ed31f64c1637	78f71ab2-f706-4df0-9279-6a5f6c9d0bdf	False	User00589	company.local
Oct 18, 2021 @ 20:09:47.624	Adobe Acrobat DC	US-DT07631	Not Licensed	Not Licensed	c2da9b1b-7b21-4d82-a2cc-111e1005c2aa	4f007ff9-0fb0-442e-9191-fe8f146fa5b0	False	User06717	company.local
Oct 18, 2021 @ 08:59:09.975	Adobe Acrobat DC	BYOD01	Not Licensed	Not Licensed	VMware-56 4d 2f 62 b0 55 f0 75-d5 1b 4f 37 8a 11 4d ea	0F29CB60-912D-4A08-93F0-1E7F33EF96D7	False	demo	DEMO

The ELK Stack Status Report Template can be used to monitor the status of your remoting and DaaS environment:



458 Total Unique Devices

457 Total Unique Users

dT - Status Report - Search - Overview

1–50 of 458

Time	Country	Access Mode	Device Type	Device Name	OS Name	Security State	Remoting Client Protocol	deviceTRUST Client	Session User Name	Session User Domain
2021-10-25T03:51:10.000Z	DE	External	IGEL (Managed)	TCDE12340251	IGEL	Protected	PCoIP	Available	Test9489	company.local
2021-10-24T17:09:11.000Z	Unknown	External	IGEL (Unmanaged)	ITCUS0847259	IGEL	Unprotected	RDP	Available	DEBB1030	Germany
2021-10-24T13:31:47.000Z	US	External	macOS	DTMACDE66769	macOS	Protected	RDP	Available	UKDW520	de.corporate.local
2021-10-23T18:43:19.000Z	DE	Internal	IGEL (Managed)	TCDE12328947	IGEL	Protected	Blast	Available	UID21089	GLOBAL
2021-10-23T17:27:17.000Z	DE	External	macOS	US-34DT34762	macOS	Unprotected	Blast	Available	DEDT6512	de.corporate.local

Step 1: Components

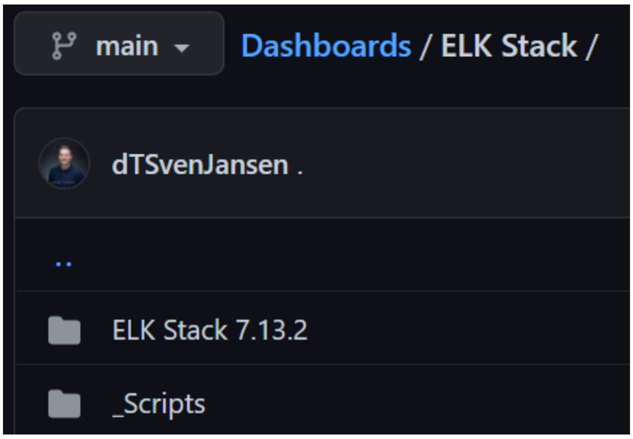
The deviceTRUST dashboards for ELK Stack consist of several components. All elements that need to be imported on the ELK Stack side can be found on GitHub. The configuration for the deviceTRUST Agent is available as a template within the deviceTRUST Console.

Step 1.1: Components - GitHub

All required components for ELK Stack can be found on our GitHub repository.

> **Note:**
>
> Saved objects for ELK are not backward compatible. Please use only versions matching or older than your system.

- Select the version number matching your ELK Stack system and download the files. You can of course also clone the whole repository if you like.

  main

  Dashboards / ELK Stack /

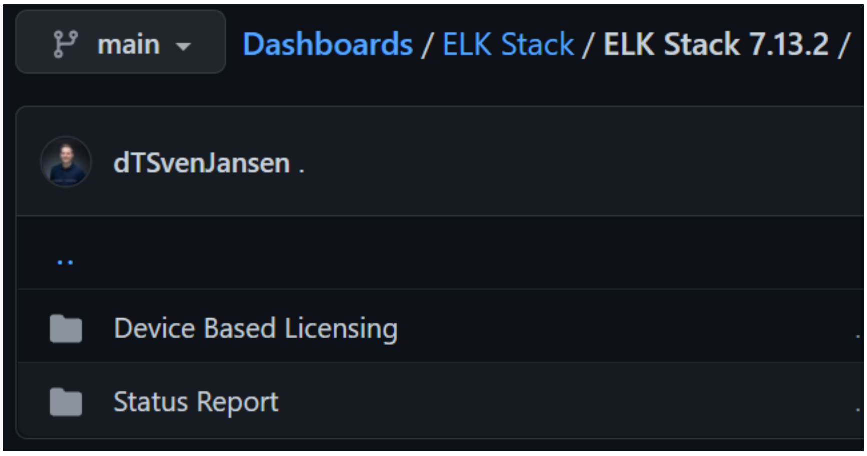
  dTSvenJansen .

  ..

  ELK Stack 7.13.2

  _Scripts

  Every version folder contains folders for each use case. Each of these use case folders contains the relevant stored object and mapping files.

  main

  Dashboards / ELK Stack / ELK Stack 7.13.2 /

  dTSvenJansen .

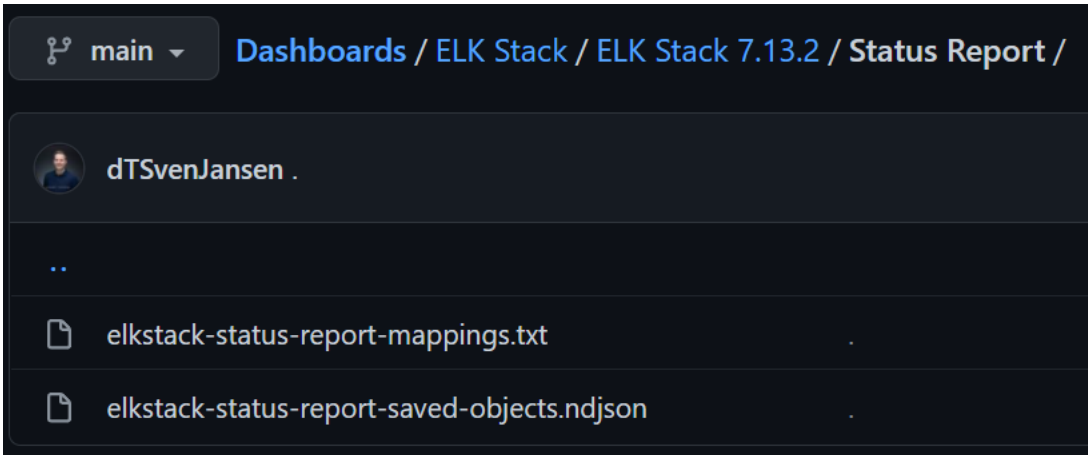
  ..

  Device Based Licensing

  Status Report

Each use case folder contains the following files:

- `elkstack-<use case>-mappings.txt` contains data definitions for the data being sent to ELK Stack index mapping.
- `elkstack-<use case>-saved-objects.ndjson` contains all objects that are required to store, search and visualize data, such as indexes, scripted fields, dashboards and searches.



main

Dashboards / ELK Stack / ELK Stack 7.13.2 / Status Report /

dTSvenJansen .

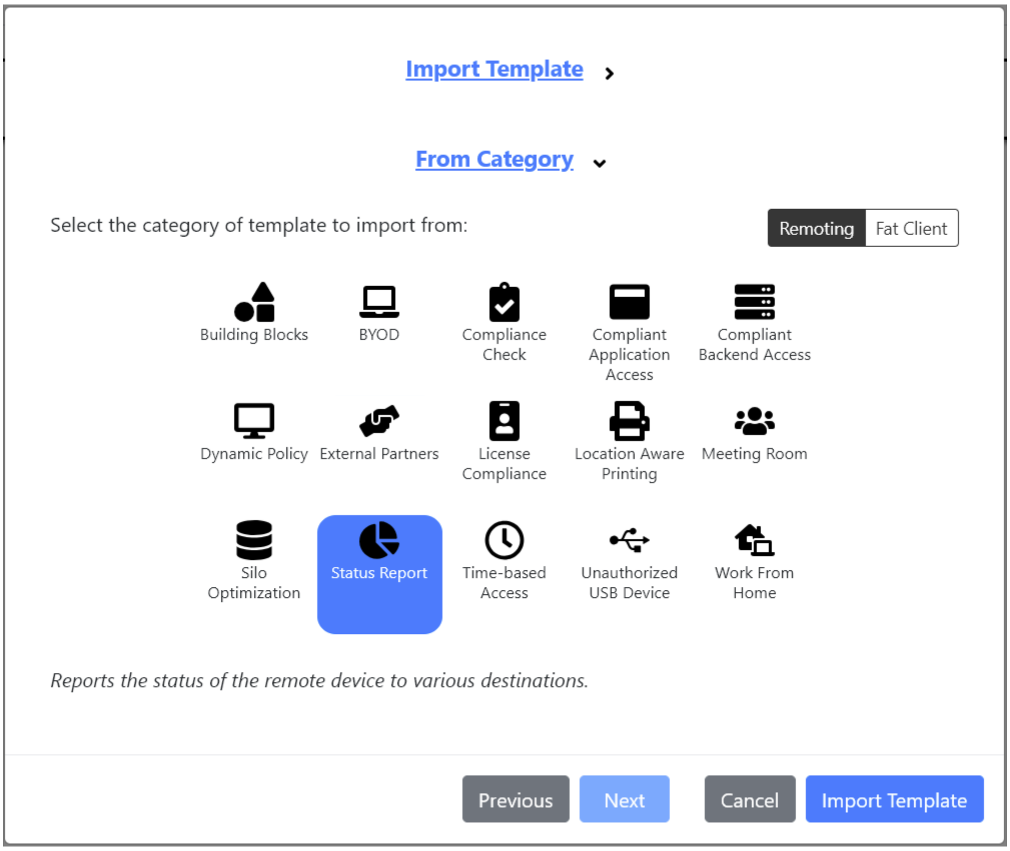
..

elkstack-status-report-mappings.txt

elkstack-status-report-saved-objects.ndjson

Step 1.2: Components - Templates

deviceTRUST must be configured to send the required data for each use case. Templates for both use cases are included in the deviceTRUST Console.



Import Template

From Category

Select the category of template to import from:

Reports the status of the remote device to various destinations.

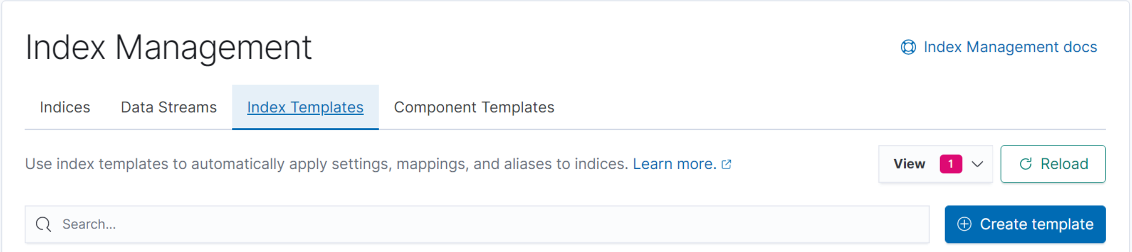
Step 2: The Device-Based Licensing Report

This part of the guide relates to `Device-Based Licensing`. It lists all steps that are required to configure the ELK Stack, and also how to configure the deviceTRUST Agent.

Step 2.1: Add Index Template to ELK Stack

The first step is to create an `Index Template`. Index Templates describe the data that is being sent to an index. They make sure that every date is treated according to its type.

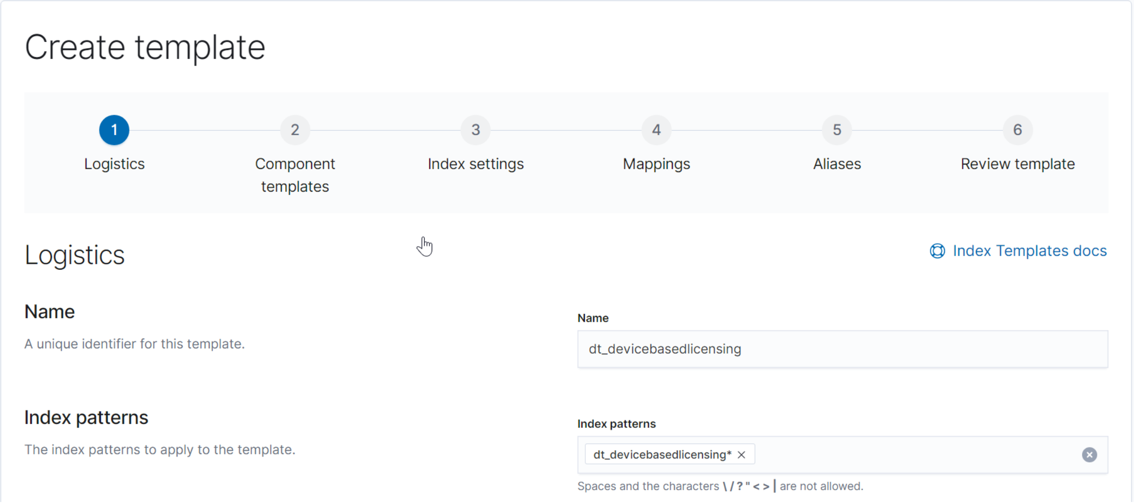
- Within the ELK Stack management console, navigate to `Menu\Stack Management\Index Management\Index Templates`.
- Click `Create Template`.



Index Management

Use index templates to automatically apply settings, mappings, and aliases to indices. Learn more.

- Set `Name` to `dt_devicebasedlicensing`.
- Set `Index Patterns` to `dt_devicebasedlicensing*`.



Create template

1 Logistics
2 Component templates
3 Index settings
4 Mappings
5 Aliases
6 Review template

Logistics

Name

A unique identifier for this template.

Name: dt_devicebasedlicensing

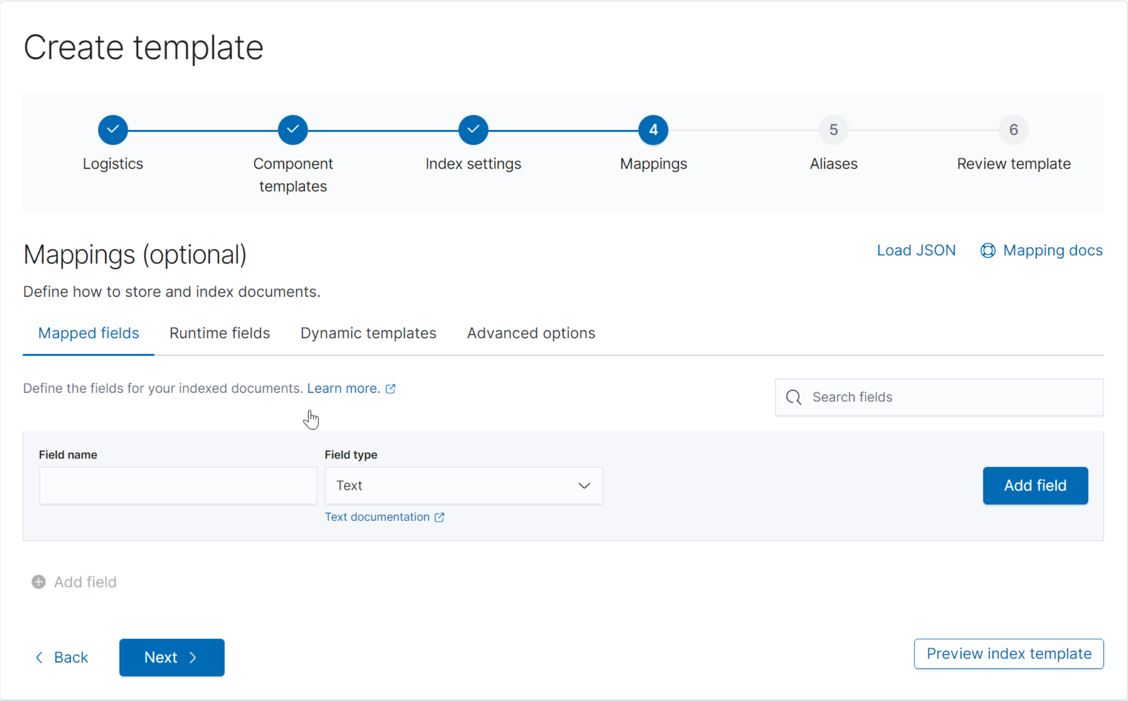
Index patterns

The index patterns to apply to the template.

Index patterns: dt_devicebasedlicensing*

Spaces and the characters \ / ? " < > | are not allowed.

- Skip all options until `Mappings`.
- Select `Load Json` to import the file `elkstack-device-based-licensing-mappings.txt` that was downloaded in Step 1.1.



Create template

Mappings (optional)

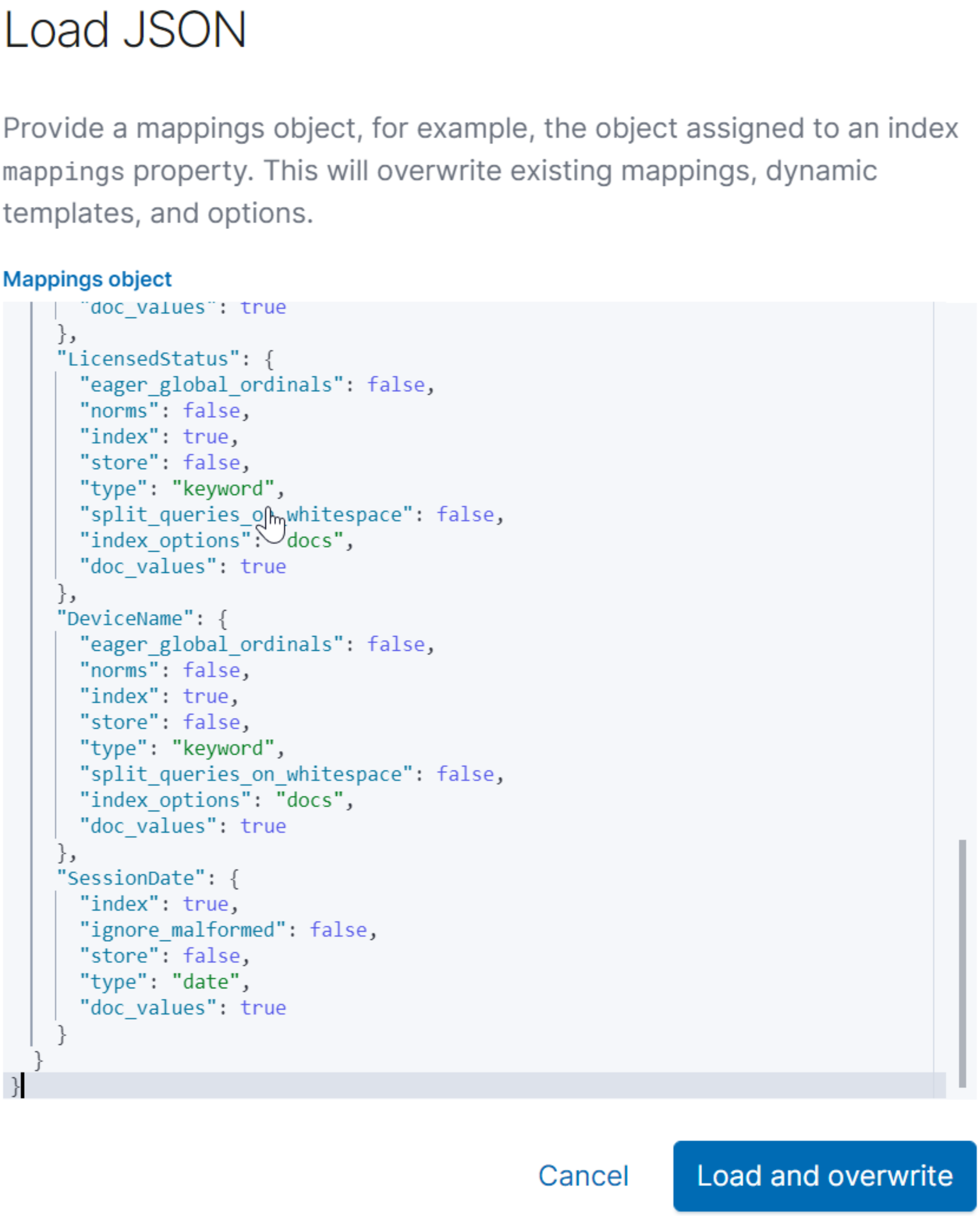
Define how to store and index documents.

Define the fields for your indexed documents. Learn more.

Field name:

Field type: Text

- Proceed with `Load and overwrite`.



Load JSON

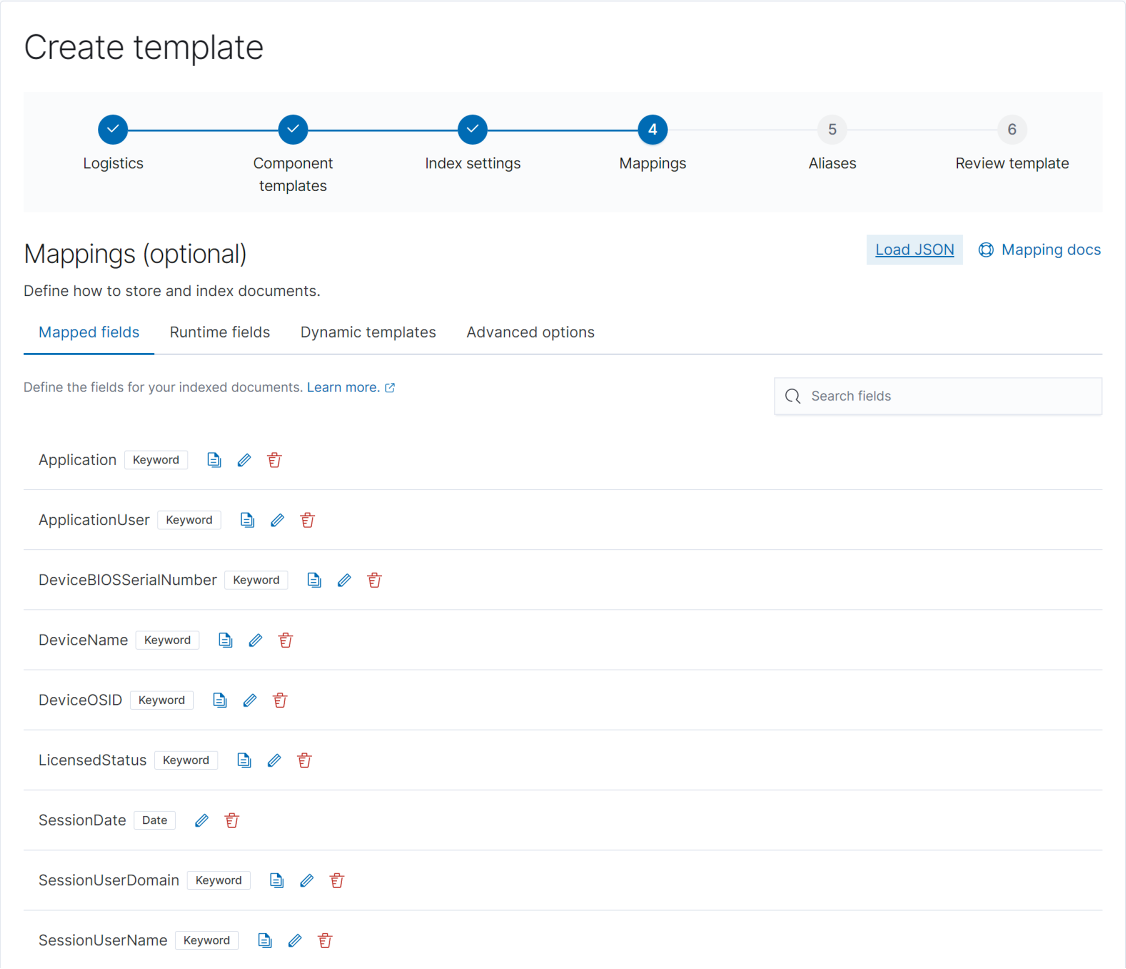
Provide a mappings object, for example, the object assigned to an index mappings property. This will overwrite existing mappings, dynamic templates, and options.

Mappings object

```
      "doc_values": true
    },
    "LicensedStatus": {
      "eager_global_ordinals": false,
      "norms": false,
      "index": true,
      "store": false,
      "type": "keyword",
      "split_queries_on_whitespace": false,
      "index_options": "docs",
      "doc_values": true
    },
    "DeviceName": {
      "eager_global_ordinals": false,
      "norms": false,
      "index": true,
      "store": false,
      "type": "keyword",
      "split_queries_on_whitespace": false,
      "index_options": "docs",
      "doc_values": true
    },
    "SessionDate": {
      "index": true,
      "ignore_malformed": false,
      "store": false,
      "type": "date",
      "doc_values": true
    }
  }
}
```

The imported mappings are displayed. Please review them carefully.

- `SessionDate` needs to be recognized as type `Date` for the report to function properly.



Create template

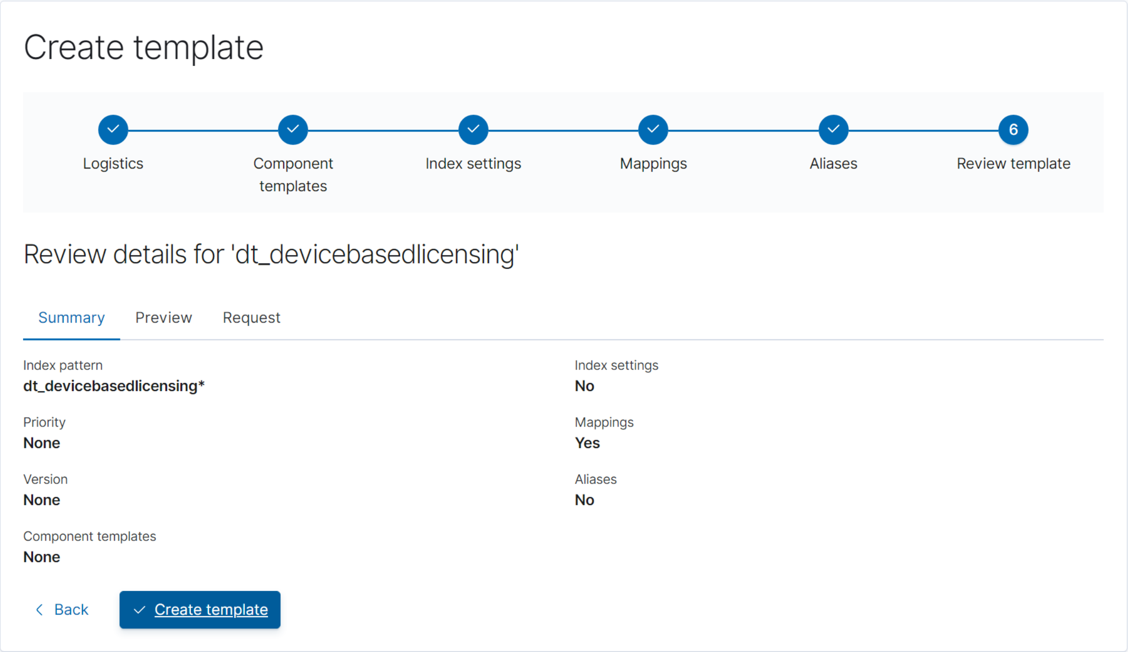
Mappings (optional)

Define how to store and index documents.

Define the fields for your indexed documents. Learn more.

Application	Keyword
ApplicationUser	Keyword
DeviceBIOSSerialNumber	Keyword
DeviceName	Keyword
DeviceOSID	Keyword
LicensedStatus	Keyword
SessionDate	Date
SessionUserDomain	Keyword
SessionUserName	Keyword

- Skip all options until `Review Template`.
- Generate the template with `Create Template`.



Create template

Review details for 'dt_devicebasedlicensing'

Index pattern
dt_devicebasedlicensing*

Priority
None

Version
None

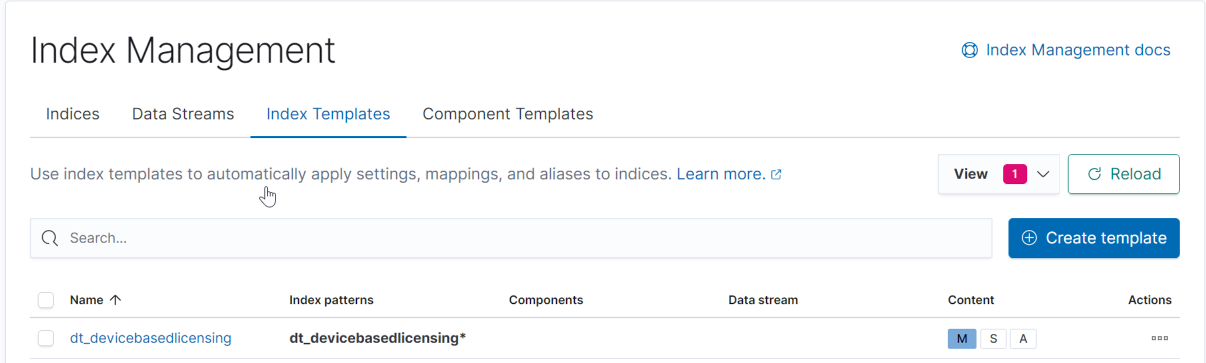
Component templates
None

Index settings
No

Mappings
Yes

Aliases
No

You'll be given an overview of the created template. A blue marked `M` in the `Content` section indicates that mappings are available.



Index Management

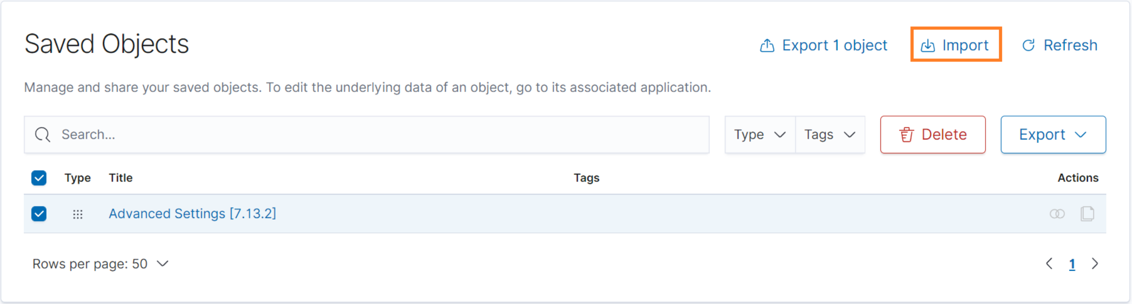
Use index templates to automatically apply settings, mappings, and aliases to indices. Learn more.

□	Name ↑	Index patterns	Components	Data stream	Content	Actions
□	dt_devicebasedlicensing	dt_devicebasedlicensing*			M S A	···

Step 2.2: Import Saved Objects to ELK Stack

All other parts of the report are to be imported as `Saved Objects`. The Saved Objects consist of Index Patterns with Scripted Fields, Visualizations and Dashboards.

- Navigate to `Menu\Stack Management\Saved Objects` in your ELK Stack management console.
- Click `Import`.



Saved Objects

Export 1 object | Import | Refresh

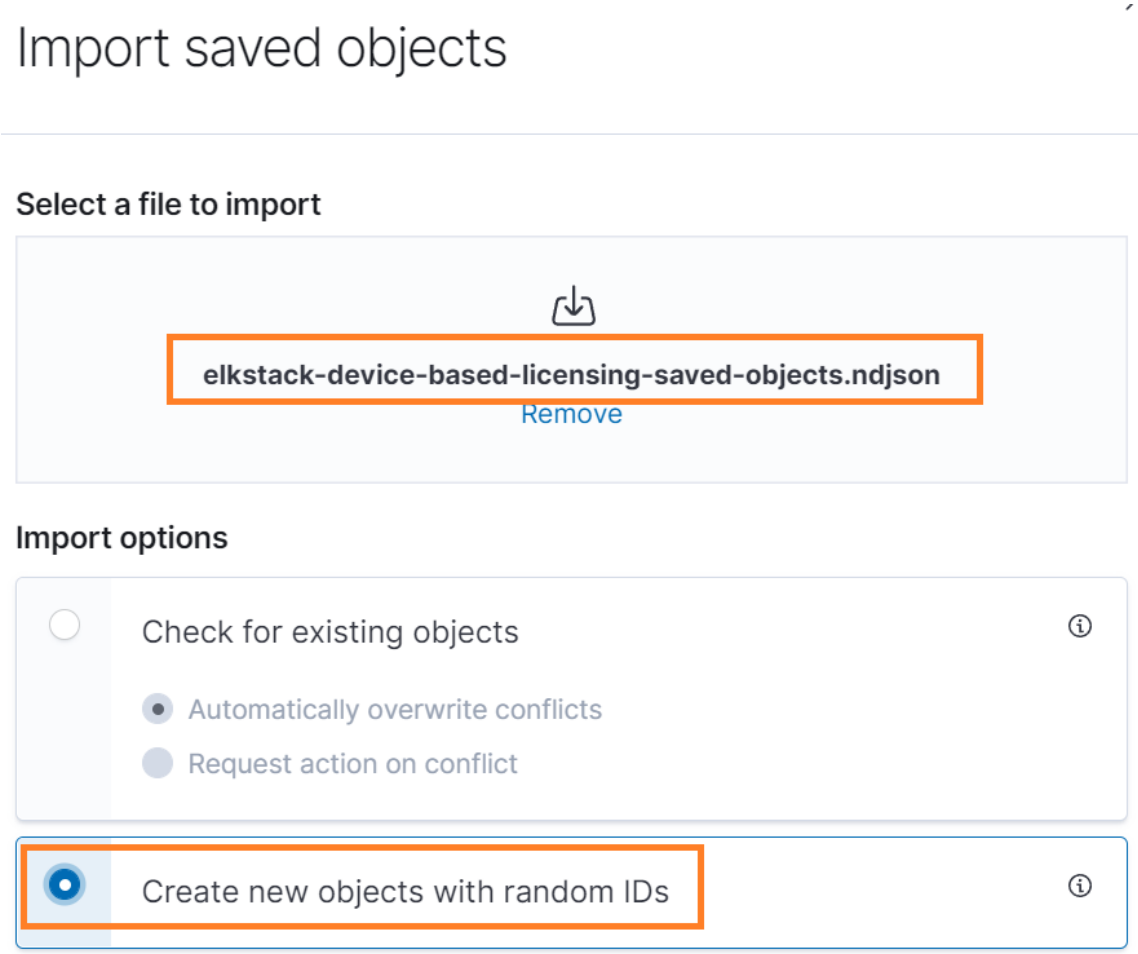
Manage and share your saved objects. To edit the underlying data of an object, go to its associated application.

Search... | Type | Tags | Delete | Export

☑	Type	Title	Tags	Actions
☑		Advanced Settings [7.13.2]		

Rows per page: 50 ‹ 1 ›

- Select the file `elkstack-device-based-licensing-saved-objects.ndjson`.
- Check `Create new objects with random IDs` to make sure no existing objects are altered.



Import saved objects

Select a file to import

elkstack-device-based-licensing-saved-objects.ndjson

Remove

Import options

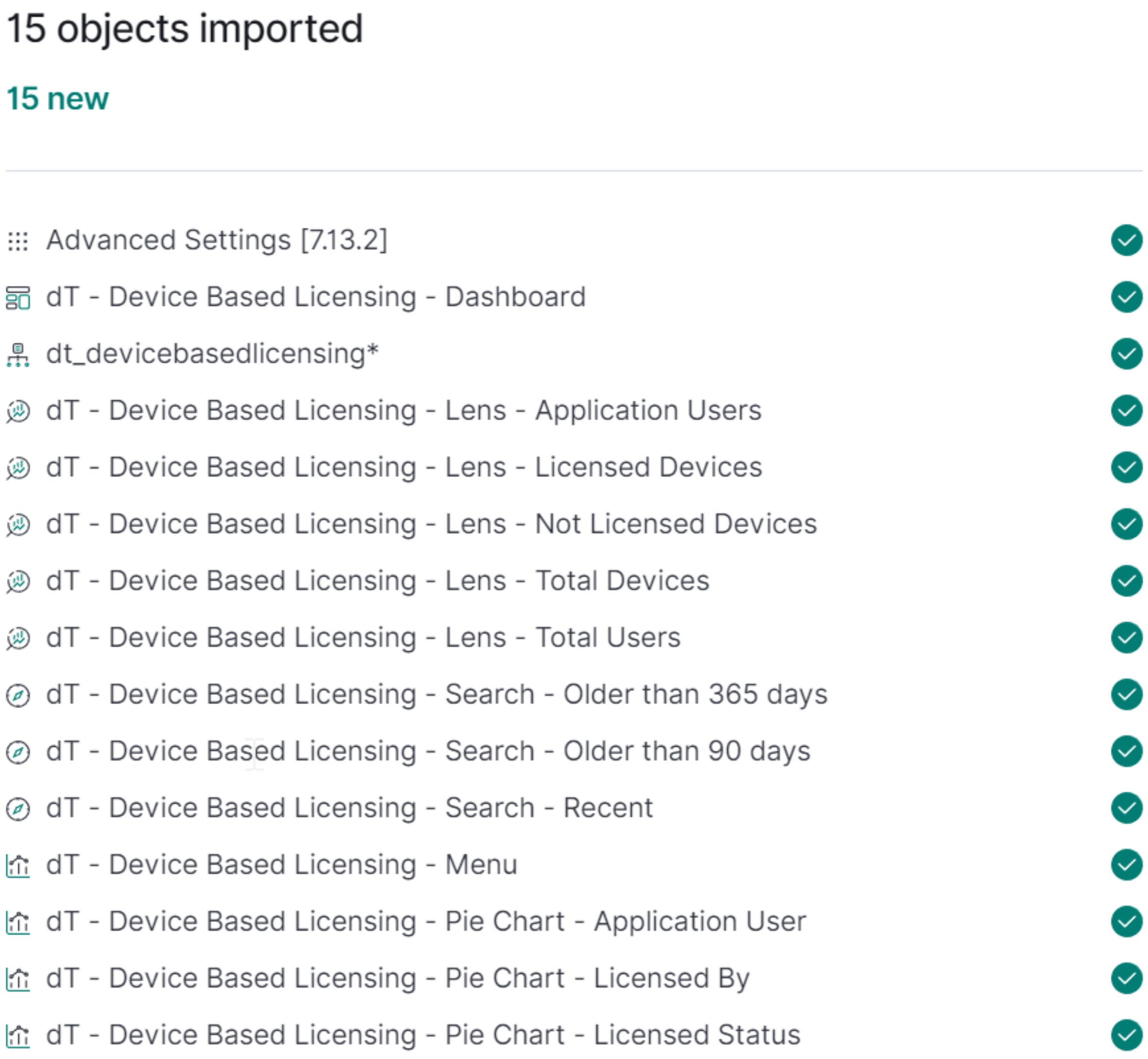
○ Check for existing objects

● Automatically overwrite conflicts

○ Request action on conflict

● Create new objects with random IDs

After importing, an overview of the imported objects will be displayed.



15 objects imported

15 new

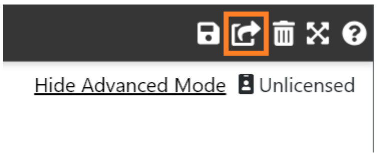
- Advanced Settings [7.13.2] ✓
- dT - Device Based Licensing - Dashboard ✓
- dt_devicebasedlicensing* ✓
- dT - Device Based Licensing - Lens - Application Users ✓
- dT - Device Based Licensing - Lens - Licensed Devices ✓
- dT - Device Based Licensing - Lens - Not Licensed Devices ✓
- dT - Device Based Licensing - Lens - Total Devices ✓
- dT - Device Based Licensing - Lens - Total Users ✓
- dT - Device Based Licensing - Search - Older than 365 days ✓
- dT - Device Based Licensing - Search - Older than 90 days ✓
- dT - Device Based Licensing - Search - Recent ✓
- dT - Device Based Licensing - Menu ✓
- dT - Device Based Licensing - Pie Chart - Application User ✓
- dT - Device Based Licensing - Pie Chart - Licensed By ✓
- dT - Device Based Licensing - Pie Chart - Licensed Status ✓

Step 2.3: Configuring deviceTRUST

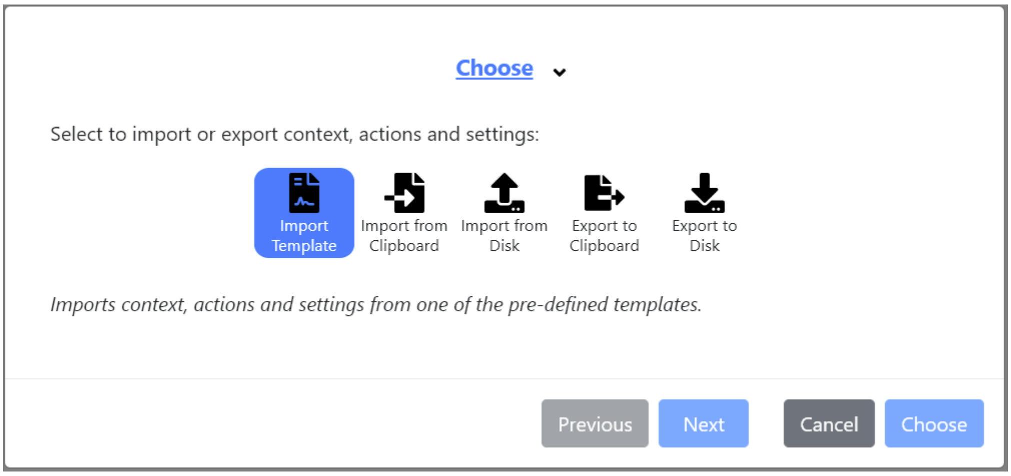
After the Index Mapping has been created and the Saved Objects are included, the ELK Stack is prepared for storing, sorting, and displaying your data.

The final step is to create the deviceTRUST configuration that will make sure all the required data is provided.

- Open the `deviceTRUST Console`.
- Click `Sharing` in the top right menu. You may need to click `Show Advanced View` if this button is not visible.



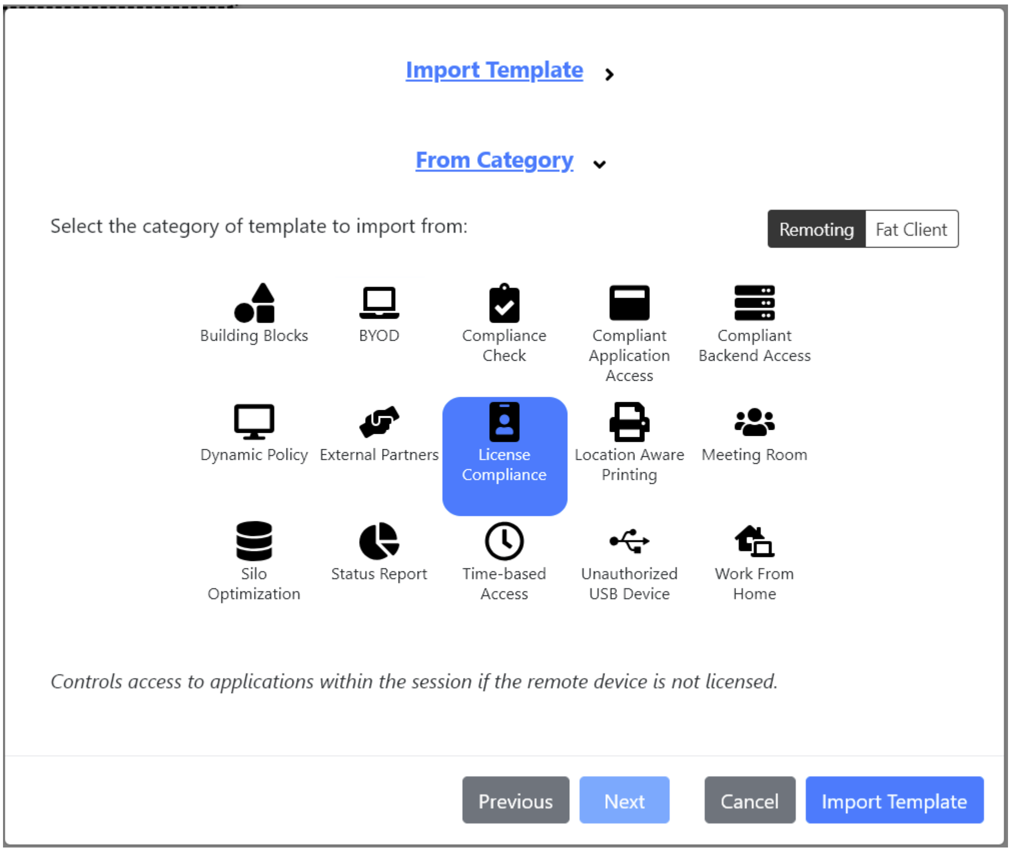
- Select `Import Template`.



Select to import or export context, actions and settings:

Imports context, actions and settings from one of the pre-defined templates.

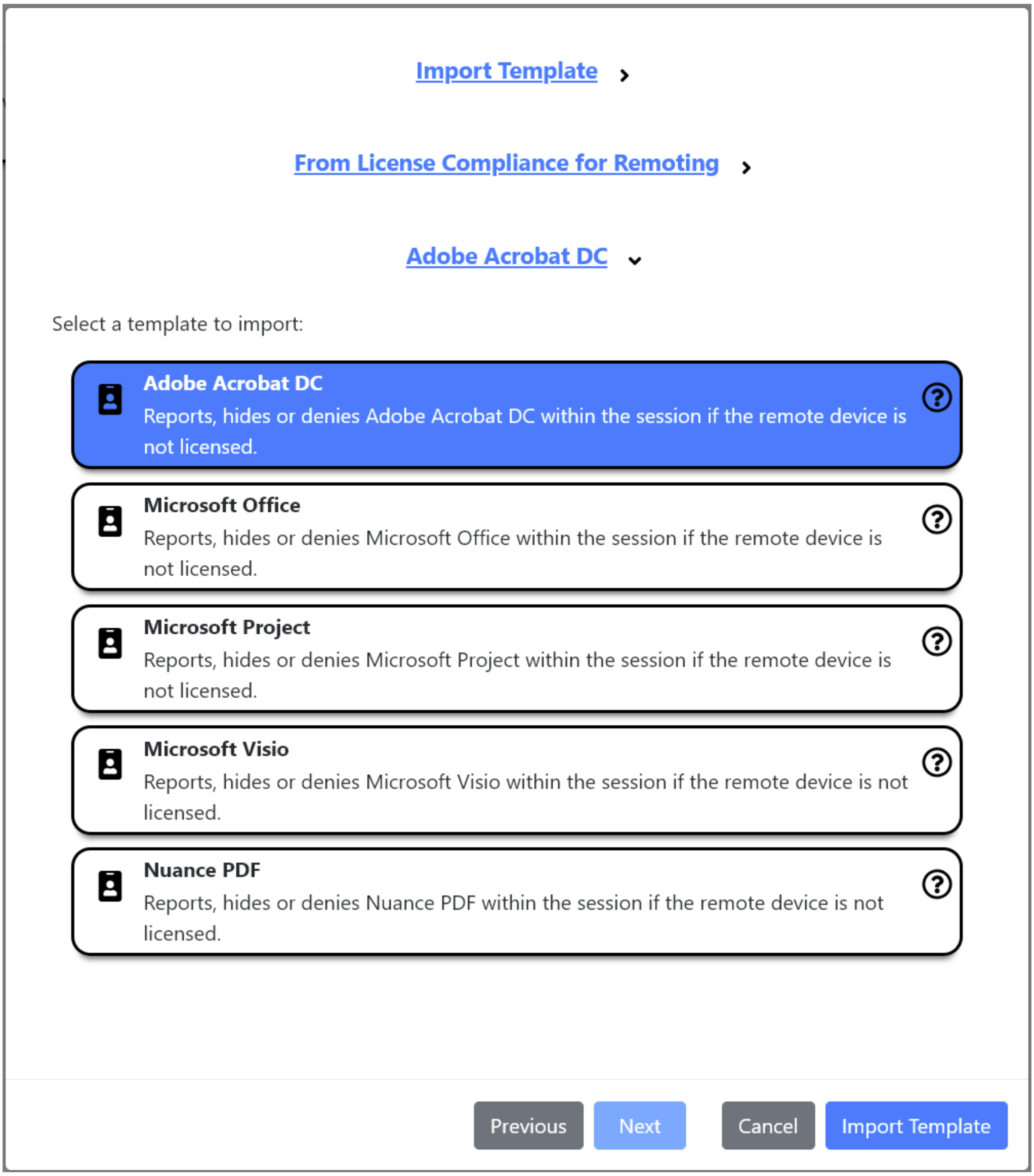
- For `Device-Based Licensing`, the template category `License Compliance` is used.



Select the category of template to import from:

Controls access to applications within the session if the remote device is not licensed.

This category contains templates for several software products. This example uses `Acrobat DC`, but can easily be customised for other applications.



Select a template to import:

Adobe Acrobat DC
Reports, hides or denies Adobe Acrobat DC within the session if the remote device is not licensed.

Microsoft Office
Reports, hides or denies Microsoft Office within the session if the remote device is not licensed.

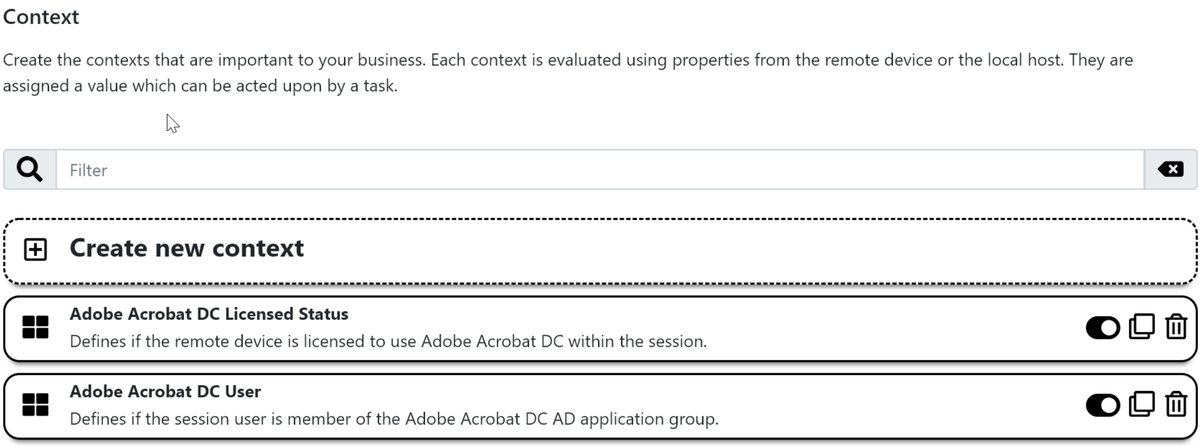
Microsoft Project
Reports, hides or denies Microsoft Project within the session if the remote device is not licensed.

Microsoft Visio
Reports, hides or denies Microsoft Visio within the session if the remote device is not licensed.

Nuance PDF
Reports, hides or denies Nuance PDF within the session if the remote device is not licensed.

Two contexts are included:

- `Adobe Acrobat DC Licensed Status` to evaluate the device's license status.
- `Adobe Acrobar DC User` to define if the accessing user shall or shall not be using the software.



Context

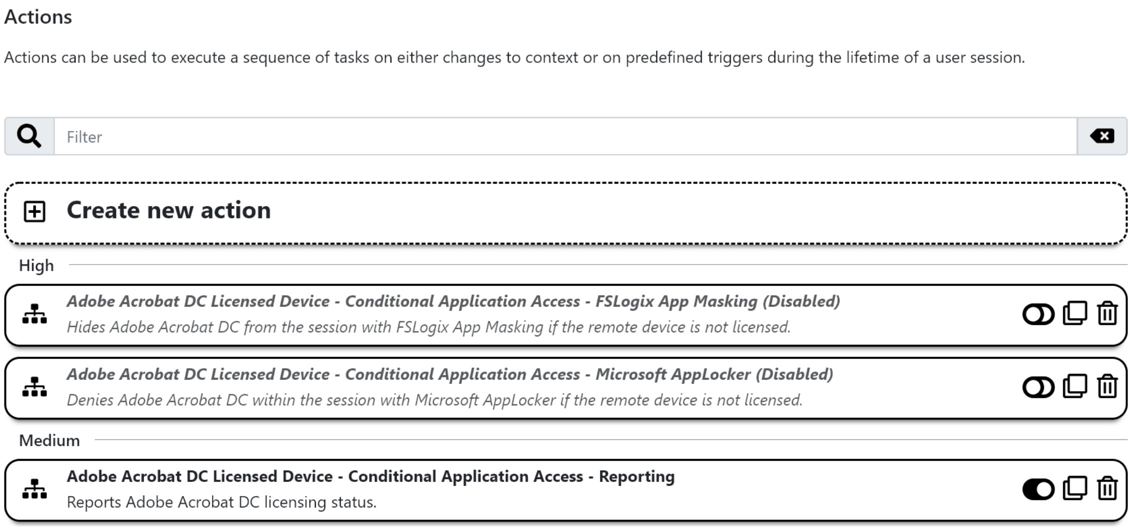
Create the contexts that are important to your business. Each context is evaluated using properties from the remote device or the local host. They are assigned a value which can be acted upon by a task.

Adobe Acrobat DC Licensed Status
Defines if the remote device is licensed to use Adobe Acrobat DC within the session.

Adobe Acrobat DC User
Defines if the session user is member of the Adobe Acrobat DC AD application group.

Three actions are included:

- `Adobe Acrobat DC Licensed Device - Conditional Application Access - FSLogix App Masking` is used for controlling access to the software using FSLogix App Masking. This action can be ignored or removed for now.
- `Adobe Acrobat DC Licensed Device - Conditional Application Access - Microsoft AppLocker` is used for controlling access to the software using Microsoft Applocker. This action can be ignored or removed for now.
- `Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting` is the only action required for reporting.



Actions

Actions can be used to execute a sequence of tasks on either changes to context or on predefined triggers during the lifetime of a user session.

High

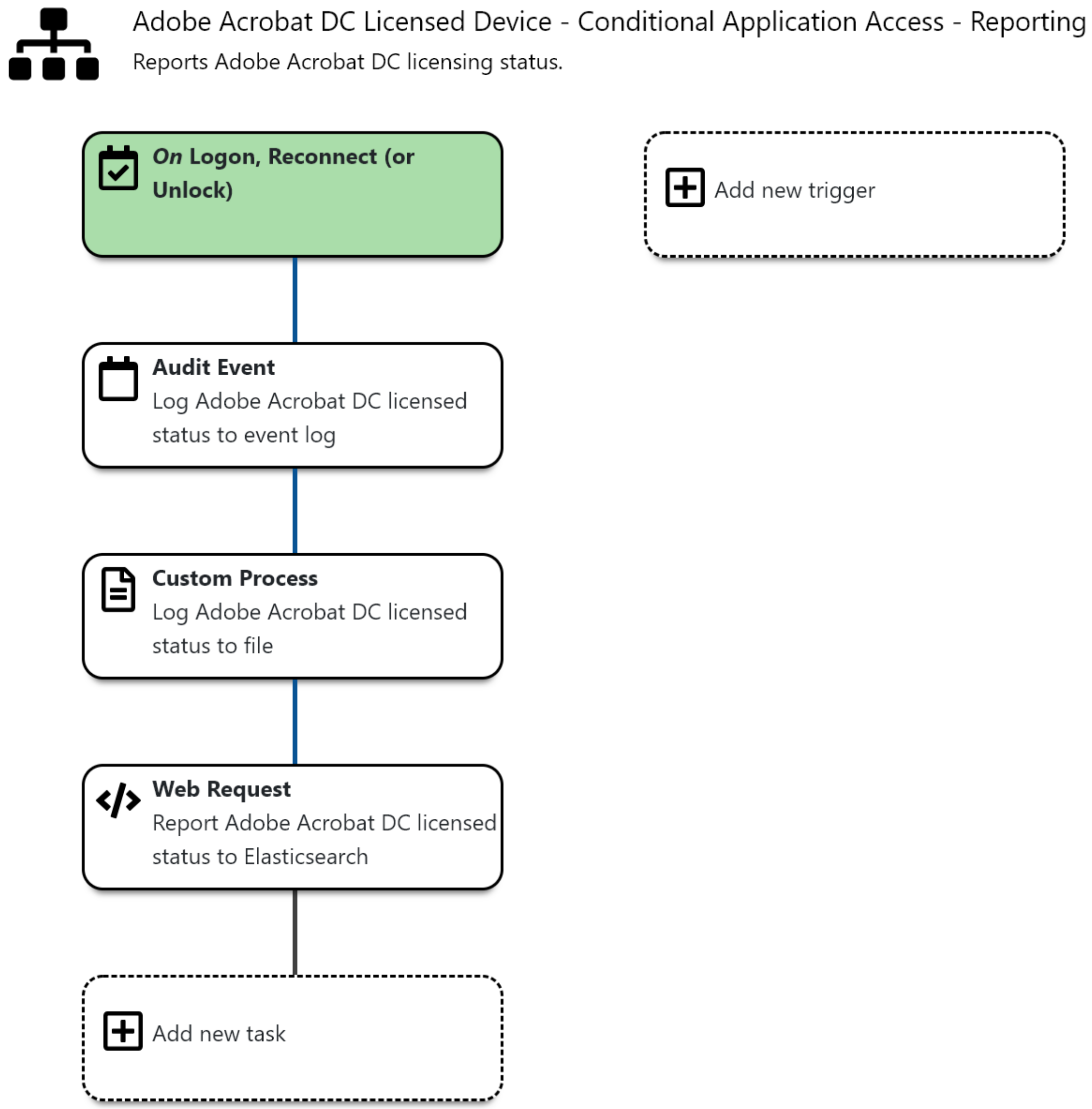
Adobe Acrobat DC Licensed Device - Conditional Application Access - FSLogix App Masking (Disabled)
Hides Adobe Acrobat DC from the session with FSLogix App Masking if the remote device is not licensed.

Adobe Acrobat DC Licensed Device - Conditional Application Access - Microsoft AppLocker (Disabled)
Denies Adobe Acrobat DC within the session with Microsoft AppLocker if the remote device is not licensed.

Medium

Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting
Reports Adobe Acrobat DC licensing status.

- The action contains multiple ways to store the data. Sending data to ELK Stack is configured by using a `Web Request` task. The `Audit Event`, `Custom Process`, as well as the `Web Request` Tasks for Splunk and Graylog can be deleted, as we are looking at ELK Stack here.



Adobe Acrobat DC Licensed Device - Conditional Application Access - Reporting
Reports Adobe Acrobat DC licensing status.

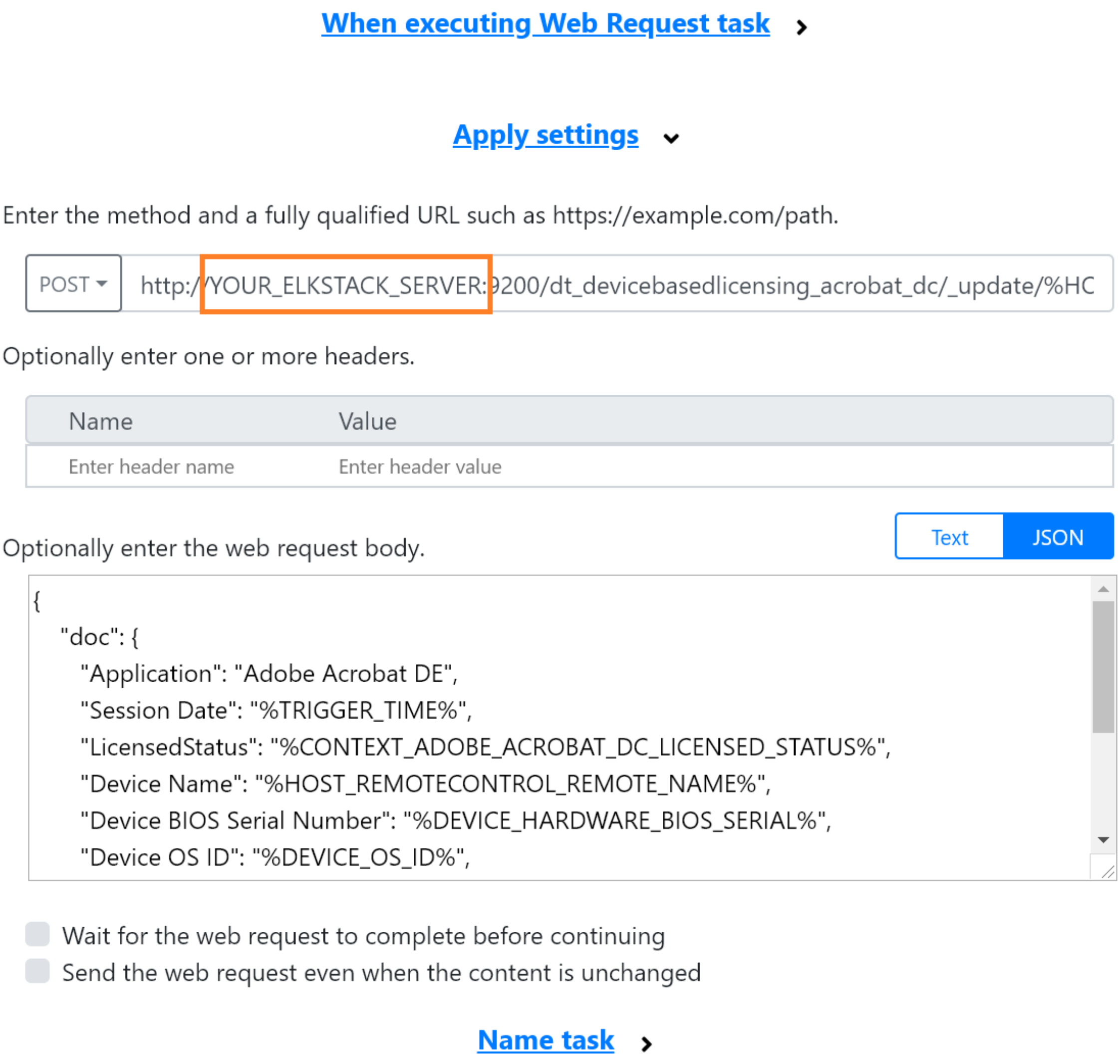
On Logon, Reconnect (or Unlock)

Audit Event
Log Adobe Acrobat DC licensed status to event log

Custom Process
Log Adobe Acrobat DC licensed status to file

Web Request
Report Adobe Acrobat DC licensed status to Elasticsearch

- The Web Request task must be edited to suit your environment. If you use a basic setup without SSL or authorization, adding `your server’s fqdn` is the only required configuration change.



When executing Web Request task ›

Apply settings ⌄

Enter the method and a fully qualified URL such as https://example.com/path.

POST ▾ http://YOUR_ELKSTACK_SERVER:9200/dt_devicebasedlicensing_acrobat_dc/_update/%HC

Optionally enter one or more headers.

Name	Value
Enter header name	Enter header value

Optionally enter the web request body.

```
{
  "doc": {
    "Application": "Adobe Acrobat DE",
    "Session Date": "%TRIGGER_TIME%",
    "LicensedStatus": "%CONTEXT_ADOBE_ACROBAT_DC_LICENSED_STATUS%",
    "Device Name": "%HOST_REMOTECONTROL_REMOTE_NAME%",
    "Device BIOS Serial Number": "%DEVICE_HARDWARE_BIOS_SERIAL%",
    "Device OS ID": "%DEVICE_OS_ID%",
```

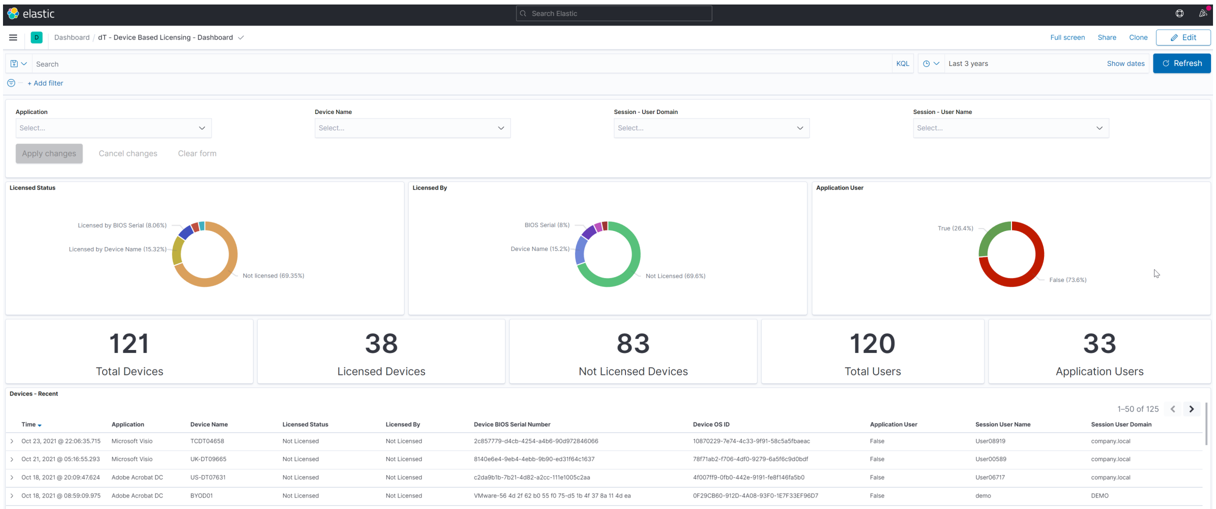
□ Wait for the web request to complete before continuing

□ Send the web request even when the content is unchanged

Name task ›

After the index template has been created, the saved objects are imported and the agent-side has been configured, the use case `Device-Based Licensing` has been implemented successfully.

deviceTRUST now sends status data about the application usage and the required hardware information to ELK Stack on every access to the remoting system. The data is presented in the created dashboards.



121 Total Devices

38 Licensed Devices

83 Not Licensed Devices

120 Total Users

33 Application Users

Devices - Recent

1–50 of 125

Time	Application	Device Name	Licensed Status	Licensed By	Device BIOS Serial Number	Device OS ID	Application User	Session User Name	Session User Domain
Oct 23, 2021 @ 22:06:35.715	Microsoft Visio	TCDT04658	Not Licensed	Not Licensed	2c857779-d4cb-4254-a4b6-90d972846066	10870229-7e74-4c33-9f91-58c5a5fbaeac	False	User08919	company.local
Oct 21, 2021 @ 05:16:55.293	Microsoft Visio	UK-DT09665	Not Licensed	Not Licensed	8140e6e4-9eb4-4ebb-9b90-ed31f64c1637	78f71ab2-f706-4df0-9279-6a5f6c9d0bdf	False	User00589	company.local
Oct 18, 2021 @ 20:09:47.624	Adobe Acrobat DC	US-DT07631	Not Licensed	Not Licensed	c2da9b1b-7b21-4d82-a2cc-111e1005c2aa	4f007ff9-0fb0-442e-9191-fe8f146fa5b0	False	User06717	company.local
Oct 18, 2021 @ 08:59:09.975	Adobe Acrobat DC	BYOD01	Not Licensed	Not Licensed	VMware-56 4d 2f 62 b0 55 f0 75-d5 1b 4f 37 8a 11 4d ea	0F29CB60-912D-4A08-93F0-1E7F33EF96D7	False	demo	DEMO

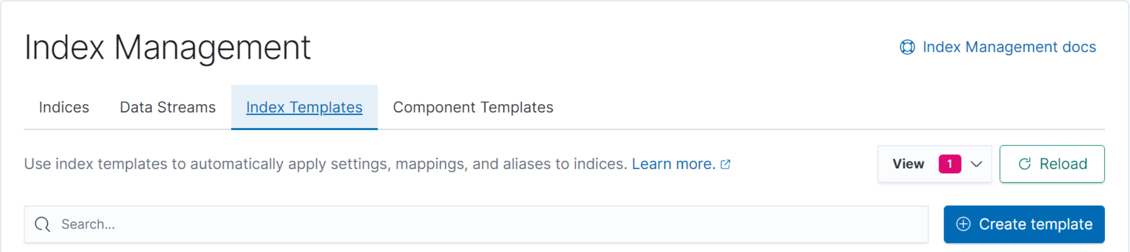
Step 3: The Status Report

This part of the guide relates to the `Status Report`. It lists all steps that are required to configure the use case on the agent-side, as well as on the ELK Stack side.

Step 3.1: Add Index Template to ELK Stack

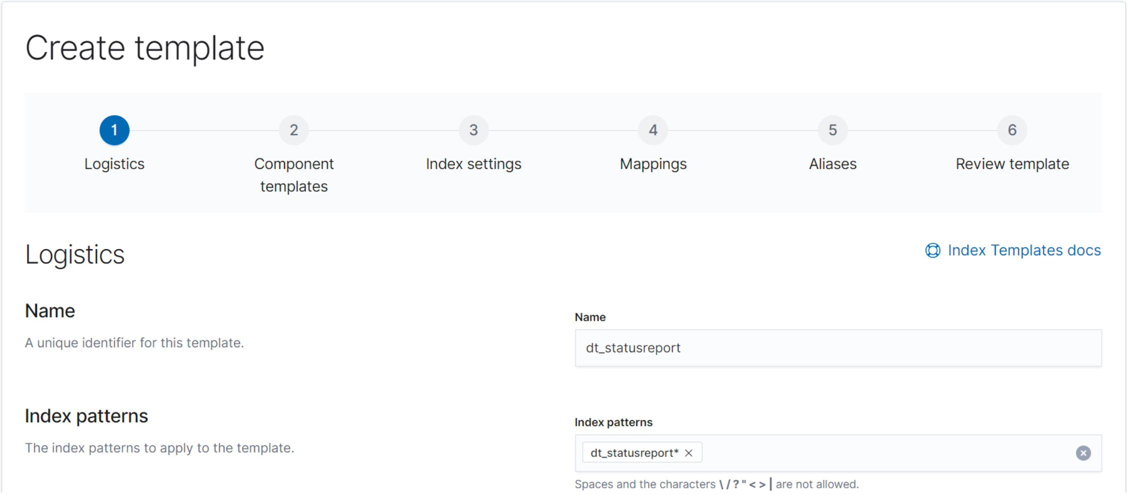
The first step is to create an `Index Template`. Index Templates describe the data that is being sent to an index. They make sure, that every date is treated according to its type.

- Within the ELK Stack management console, navigate to `Menu\Stack Management\Index Management\Index Templates`.
- Click `Create Template`.



Use index templates to automatically apply settings, mappings, and aliases to indices. Learn more.

- Set `Name` to `dt_statusreport`.
- Set `Index Patterns` to `dt_statusreport*`.



Create template

1 Logistics — 2 Component templates — 3 Index settings — 4 Mappings — 5 Aliases — 6 Review template

Logistics

Name

A unique identifier for this template.

Name: dt_statusreport

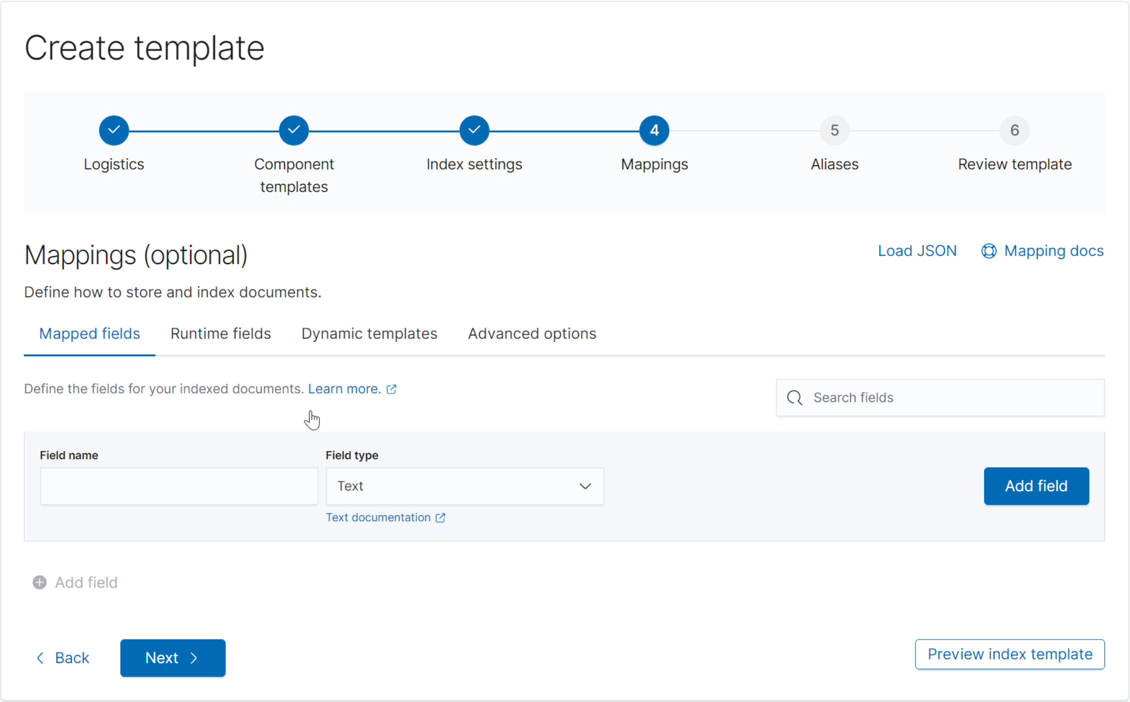
Index patterns

The index patterns to apply to the template.

Index patterns: dt_statusreport*

Spaces and the characters \ / ? " < > | are not allowed.

- Skip all options until `Mappings`.
- Select `Load Json` to import the `elkstack-status-report-mappings.txt` that was downloaded in Step 1.1.



Create template

Logistics — Component templates — Index settings — 4 Mappings — 5 Aliases — 6 Review template

Mappings (optional)

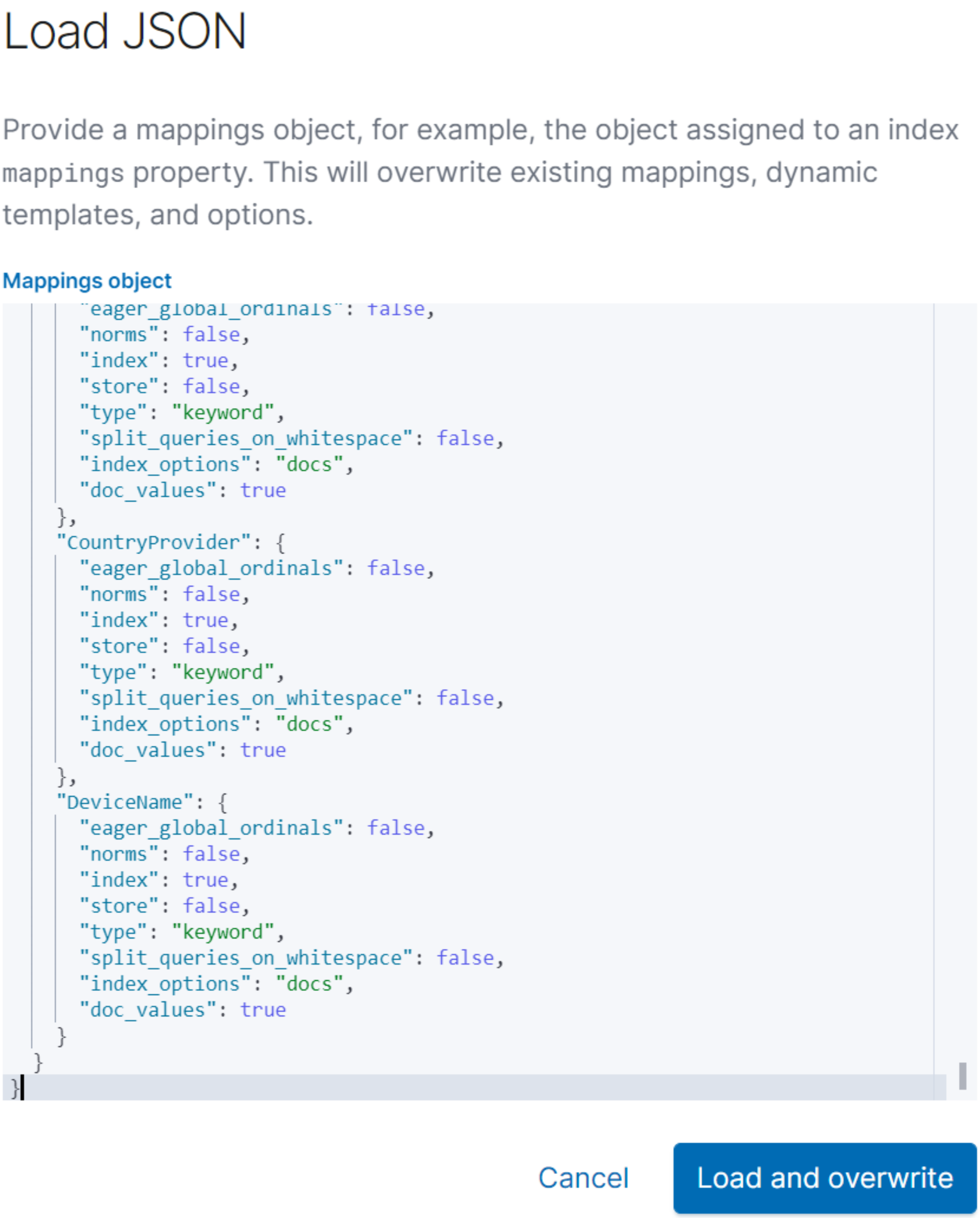
Define how to store and index documents.

Define the fields for your indexed documents. Learn more.

Field name:

Field type: Text

- Proceed with `Load and overwrite`.



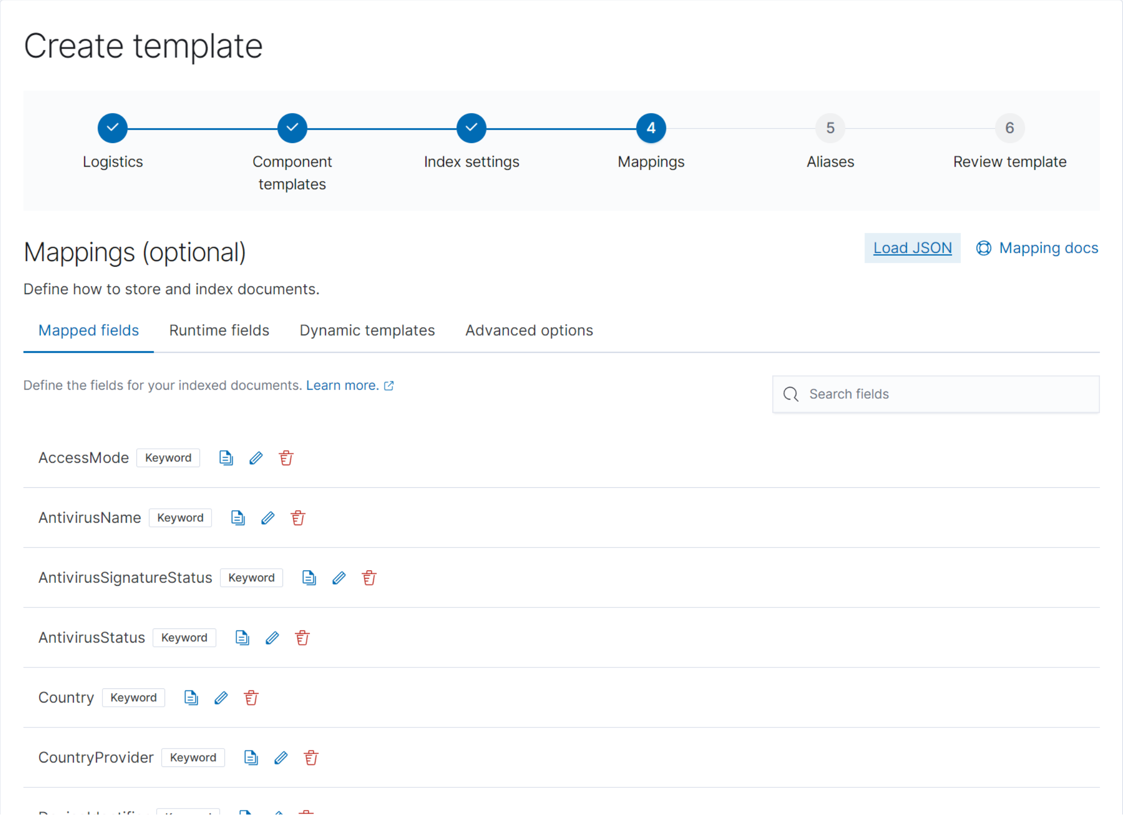
Load JSON

Provide a mappings object, for example, the object assigned to an index `mappings` property. This will overwrite existing mappings, dynamic templates, and options.

Mappings object

```
      "eager_global_ordinals": false,
      "norms": false,
      "index": true,
      "store": false,
      "type": "keyword",
      "split_queries_on_whitespace": false,
      "index_options": "docs",
      "doc_values": true
    },
    "CountryProvider": {
      "eager_global_ordinals": false,
      "norms": false,
      "index": true,
      "store": false,
      "type": "keyword",
      "split_queries_on_whitespace": false,
      "index_options": "docs",
      "doc_values": true
    },
    "DeviceName": {
      "eager_global_ordinals": false,
      "norms": false,
      "index": true,
      "store": false,
      "type": "keyword",
      "split_queries_on_whitespace": false,
      "index_options": "docs",
      "doc_values": true
    }
  }
}
```

- The imported mappings are displayed. Please review them carefully. `Session Date`, `Anti-Virus Timestamp` and `Hardware BIOS Release Date` need to be recognized as type `Date` for the report to function properly.



Create template

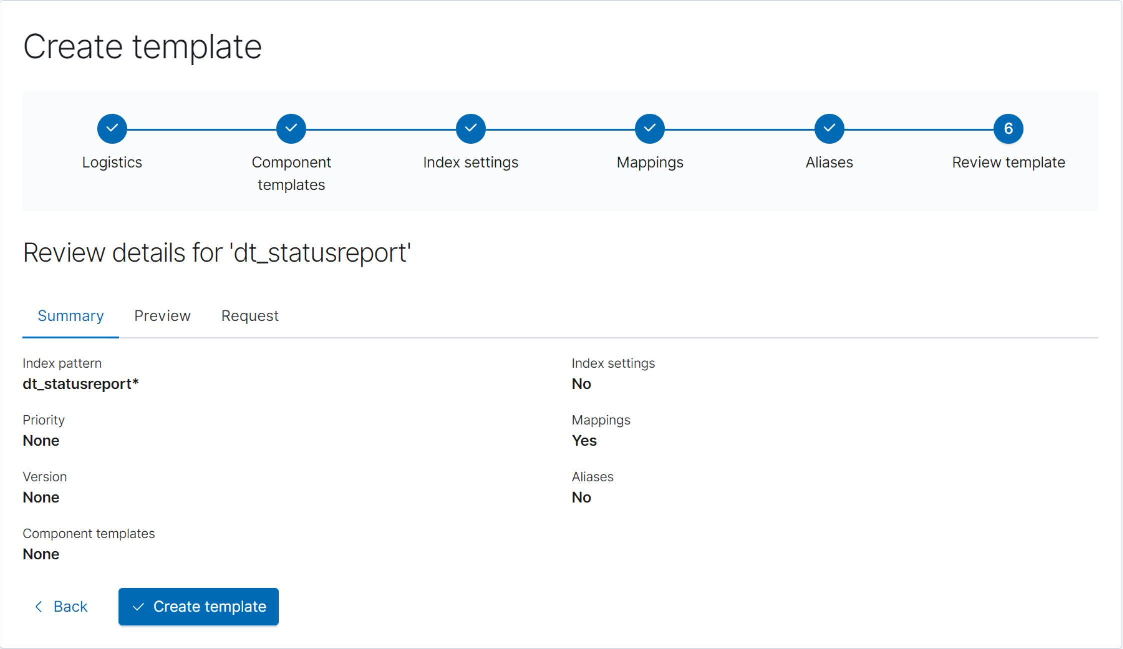
Mappings (optional)

Define how to store and index documents.

Define the fields for your indexed documents. Learn more.

AccessMode	Keyword
AntivirusName	Keyword
AntivirusSignatureStatus	Keyword
AntivirusStatus	Keyword
Country	Keyword
CountryProvider	Keyword

- Skip all options until `Review Template`.
- Generate the template with `Create Template`.



Create template

Review details for 'dt_statusreport'

Index pattern
dt_statusreport*

Priority
None

Version
None

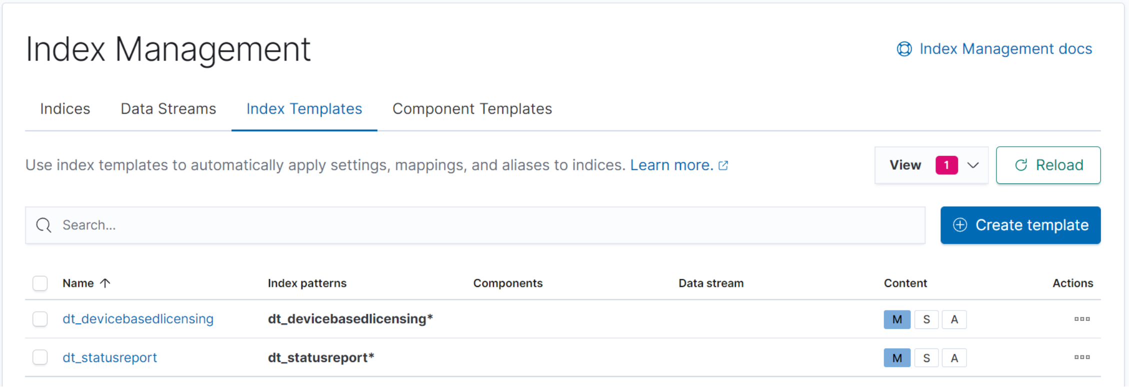
Component templates
None

Index settings
No

Mappings
Yes

Aliases
No

- You'll be given an overview of the created template. A blue marked `M` in the `Content` section indicates, that mappings are available.



Index Management

Index Management docs

Indices | Data Streams | Index Templates | Component Templates

Use index templates to automatically apply settings, mappings, and aliases to indices. Learn more.

View 1 | Reload

Search...

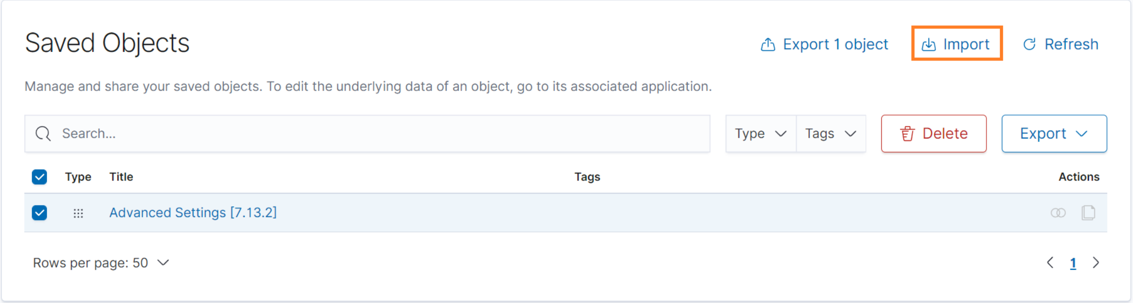
Create template

Name	Index patterns	Components	Data stream	Content	Actions
dt_devicebasedlicensing	dt_devicebasedlicensing*			M S A	···
dt_statusreport	dt_statusreport*			M S A	···

Step 3.2: Import Saved Objects to ELK Stack

All other parts of the report are to be imported as `Saved Objects`. The Saved Objects consist of Index Patterns with Scripted Fields, Visualizations and Dashboards.

- Navigate to `Menu\Stack Management\Saved Objects` in your ELK Stack management console.
- Click `Import`.



Saved Objects

Export 1 object | Import | Refresh

Manage and share your saved objects. To edit the underlying data of an object, go to its associated application.

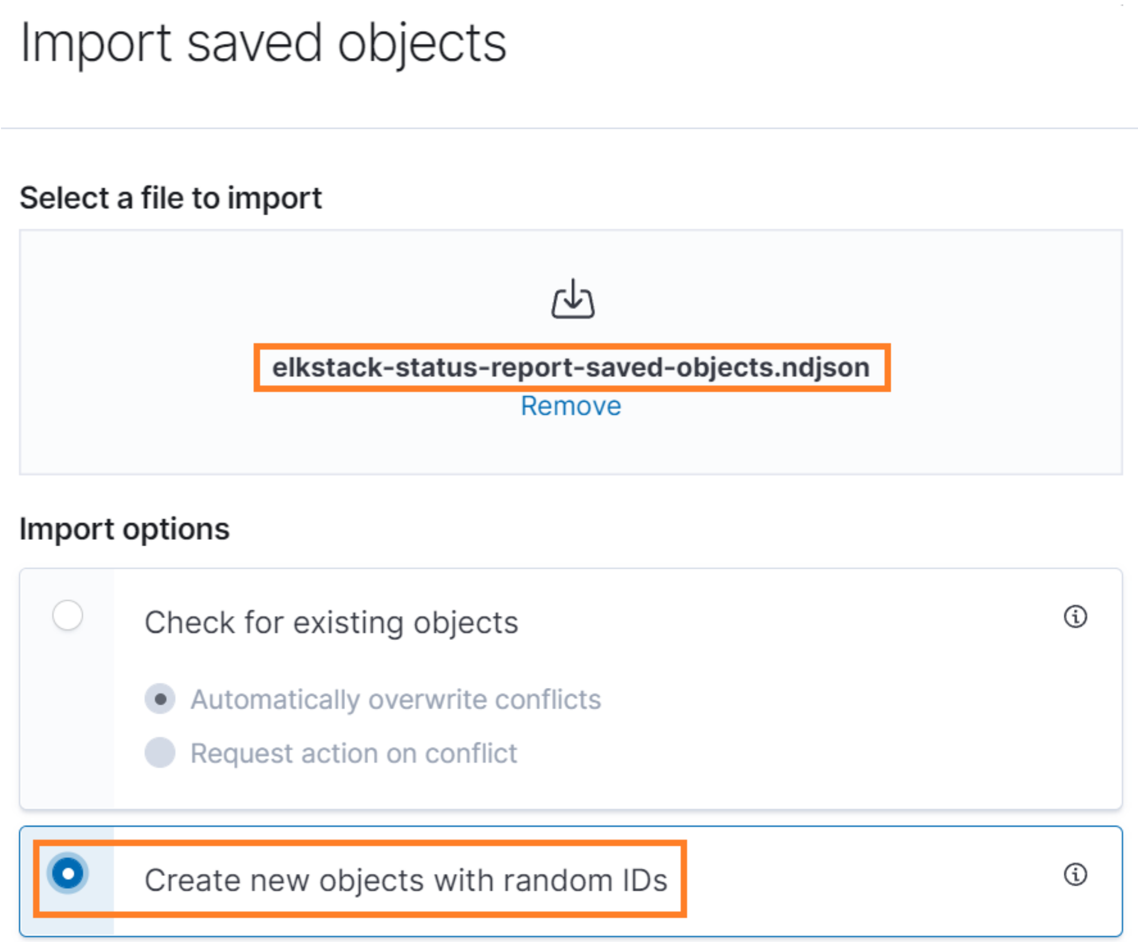
Search... | Type | Tags | Delete | Export

☑	Type	Title	Tags	Actions
☑		Advanced Settings [7.13.2]		

Rows per page: 50

< 1 >

- Select the file `elkstack-status-report-saved-objects.ndjson`.
- Check `Create new objects with random IDs` to make sure no existing objects are altered.



Import saved objects

Select a file to import

elkstack-status-report-saved-objects.ndjson

Import options

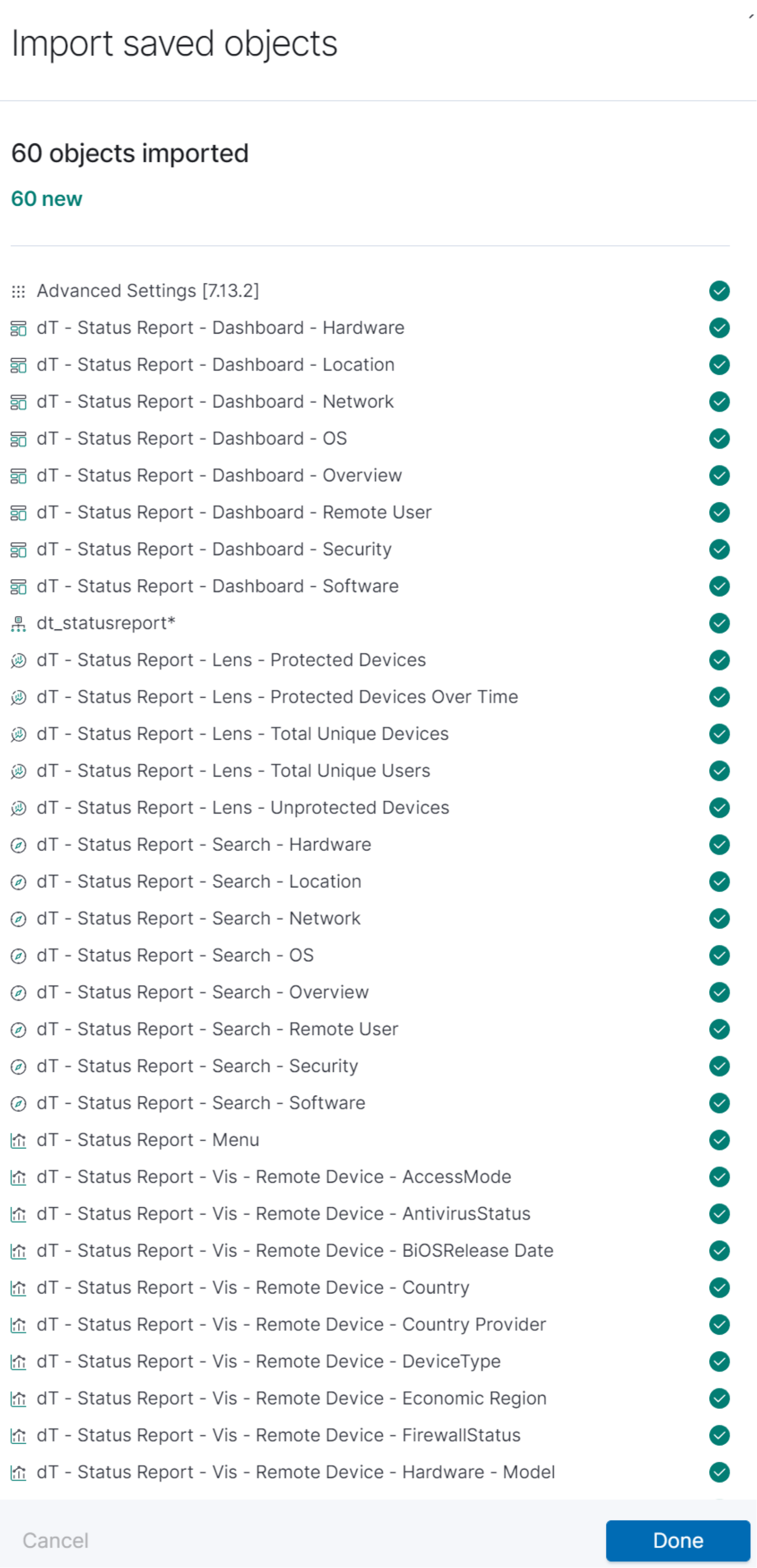
○ Check for existing objects

● Automatically overwrite conflicts

○ Request action on conflict

● Create new objects with random IDs

After importing, an overview of the imported objects will be displayed.



Import saved objects

60 objects imported

60 new

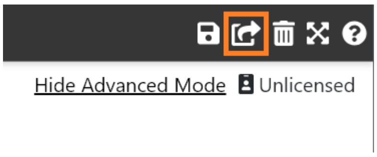
- Advanced Settings [7.13.2] ✓
- dT - Status Report - Dashboard - Hardware ✓
- dT - Status Report - Dashboard - Location ✓
- dT - Status Report - Dashboard - Network ✓
- dT - Status Report - Dashboard - OS ✓
- dT - Status Report - Dashboard - Overview ✓
- dT - Status Report - Dashboard - Remote User ✓
- dT - Status Report - Dashboard - Security ✓
- dT - Status Report - Dashboard - Software ✓
- dt_statusreport* ✓
- dT - Status Report - Lens - Protected Devices ✓
- dT - Status Report - Lens - Protected Devices Over Time ✓
- dT - Status Report - Lens - Total Unique Devices ✓
- dT - Status Report - Lens - Total Unique Users ✓
- dT - Status Report - Lens - Unprotected Devices ✓
- dT - Status Report - Search - Hardware ✓
- dT - Status Report - Search - Location ✓
- dT - Status Report - Search - Network ✓
- dT - Status Report - Search - OS ✓
- dT - Status Report - Search - Overview ✓
- dT - Status Report - Search - Remote User ✓
- dT - Status Report - Search - Security ✓
- dT - Status Report - Search - Software ✓
- dT - Status Report - Menu ✓
- dT - Status Report - Vis - Remote Device - AccessMode ✓
- dT - Status Report - Vis - Remote Device - AntivirusStatus ✓
- dT - Status Report - Vis - Remote Device - BiOSRelease Date ✓
- dT - Status Report - Vis - Remote Device - Country ✓
- dT - Status Report - Vis - Remote Device - Country Provider ✓
- dT - Status Report - Vis - Remote Device - DeviceType ✓
- dT - Status Report - Vis - Remote Device - Economic Region ✓
- dT - Status Report - Vis - Remote Device - FirewallStatus ✓
- dT - Status Report - Vis - Remote Device - Hardware - Model ✓

Step 3.3: Configuring deviceTRUST

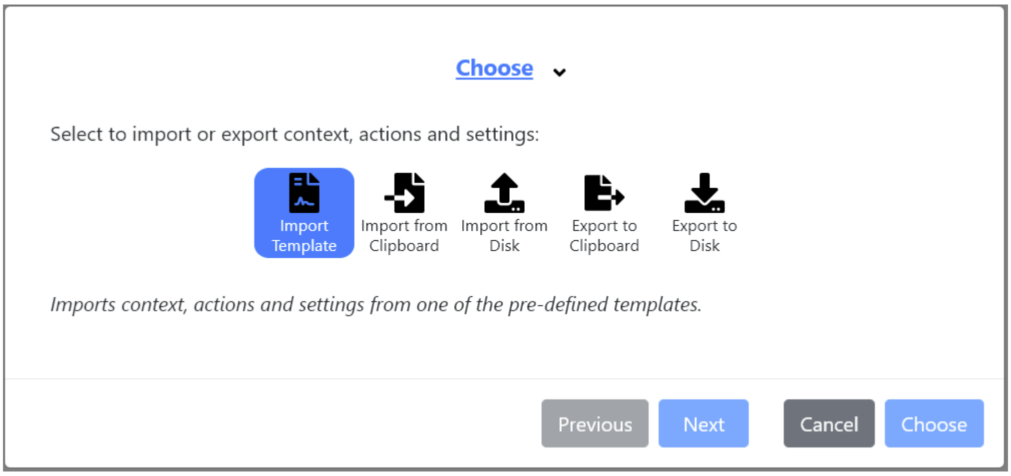
After the Index Mapping has been created, the Saved Objects are included the index has been edited and the agent-side has been configured, the ELK Stack is prepared for storing, sorting, and displaying your data.

The last step is to create the deviceTRUST configuration, that will make sure all required data is provided.

- Open the `deviceTRUST Console`.
- Click `Sharing` in the top right menu. You may need to click `Show Advanced View` if this button is not visible.



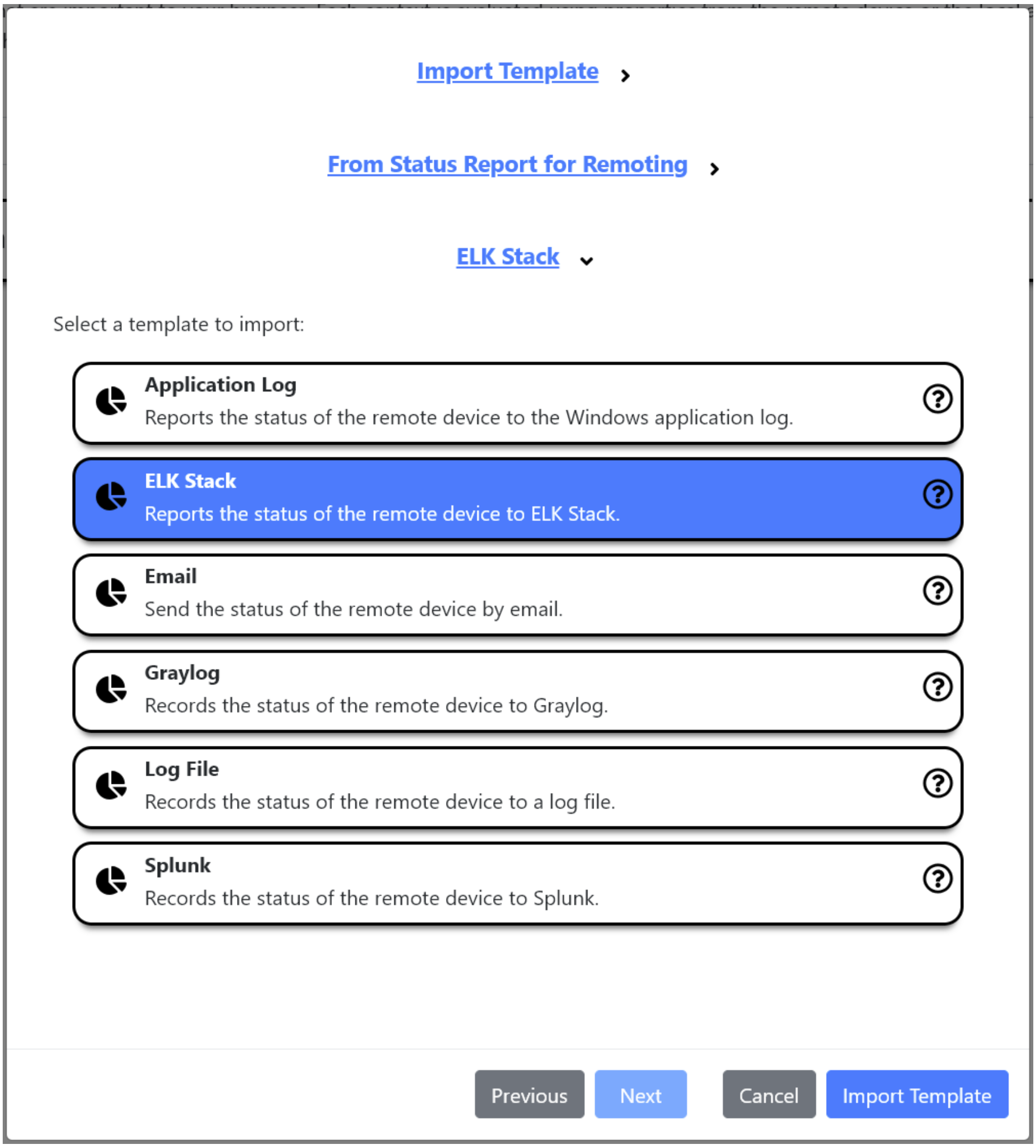
- Select `Import Template`.



Select to import or export context, actions and settings:

Imports context, actions and settings from one of the pre-defined templates.

- For `Status Report`, the template category `Status Report` is used.
- This category contains templates for several ways of storing data. Choose `ELK Stack`.



Import Template ›

From Status Report for Remoting ›

ELK Stack ⌄

Select a template to import:

Application Log
Reports the status of the remote device to the Windows application log.

ELK Stack
Reports the status of the remote device to ELK Stack.

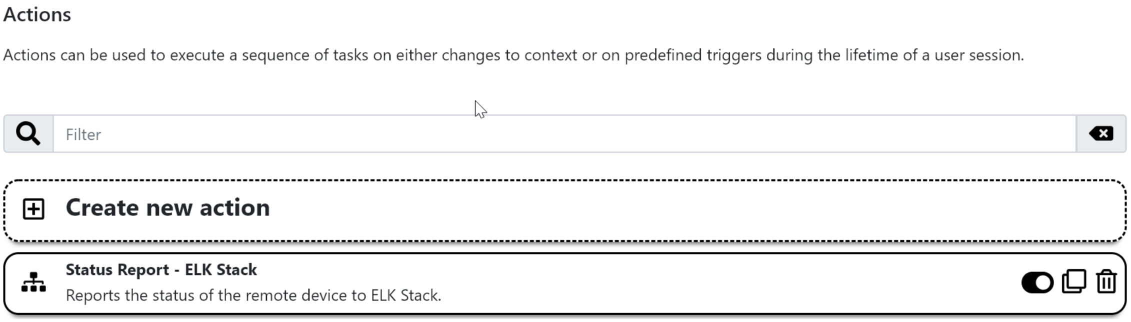
Email
Send the status of the remote device by email.

Graylog
Records the status of the remote device to Graylog.

Log File
Records the status of the remote device to a log file.

Splunk
Records the status of the remote device to Splunk.

- The imported template consists of 50 contexts and one action.
- The Action `Status Report – ELK Stack` collects all relevant data and sends them over to the ELK Stack.



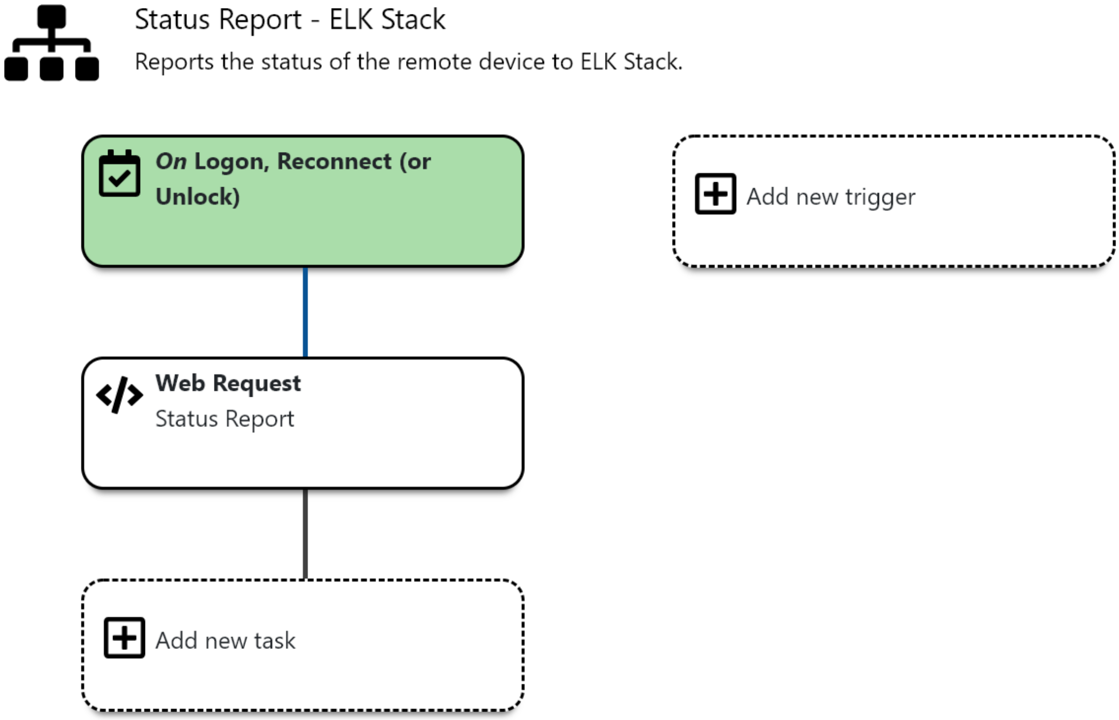
Actions

Actions can be used to execute a sequence of tasks on either changes to context or on predefined triggers during the lifetime of a user session.

Status Report - ELK Stack

Reports the status of the remote device to ELK Stack.

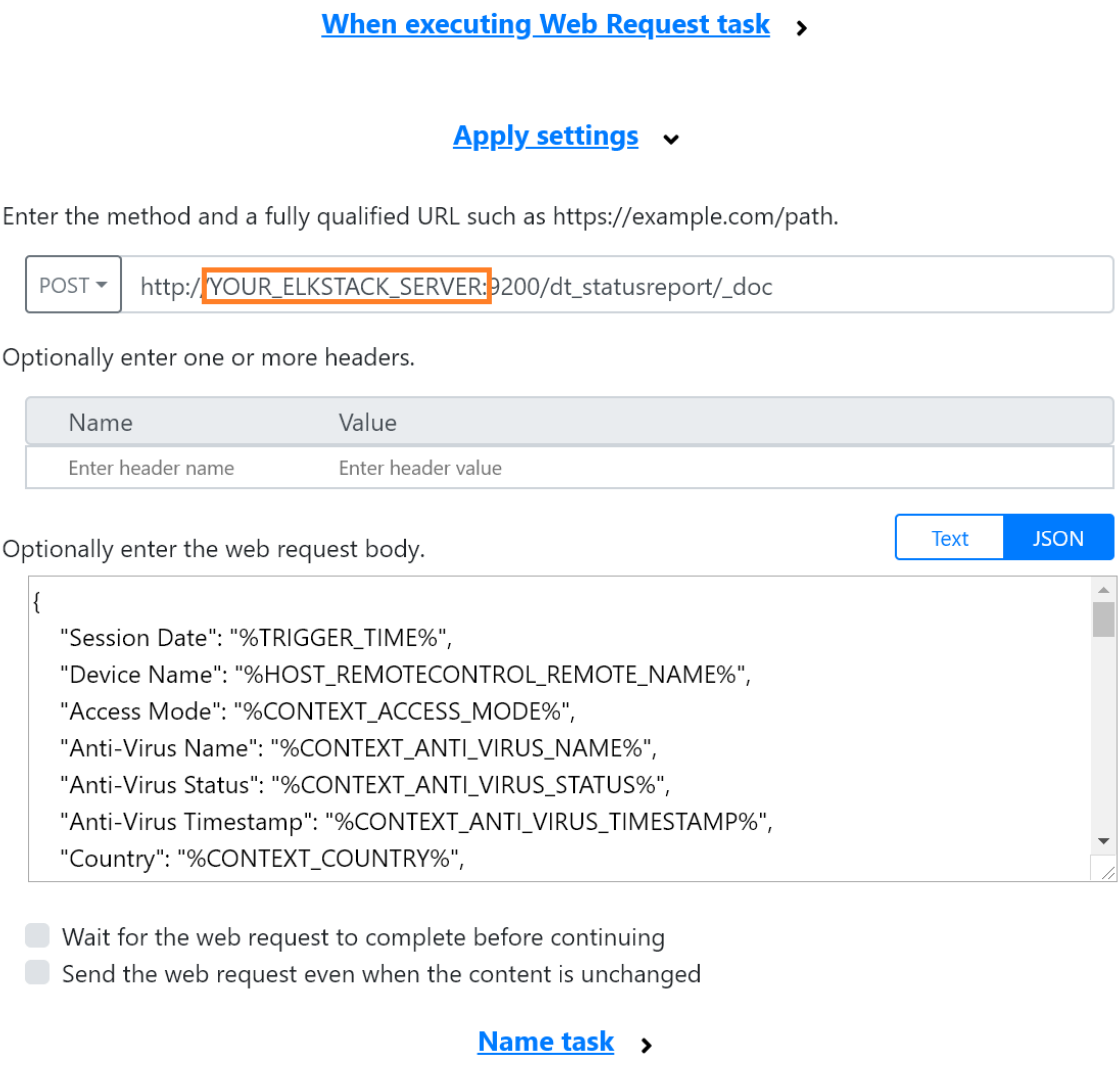
- Sending data to ELK Stack is configured by using a `Web Request` task.



Status Report - ELK Stack

Reports the status of the remote device to ELK Stack.

- The Web Request task must be edited to suit your environment. If you use a basic setup without SSL or authorization, simply adding your `server's fqdn` will do.



Enter the method and a fully qualified URL such as https://example.com/path.

POST ▾ http://YOUR_ELKSTACK_SERVER:9200/dt_statusreport/_doc

Optionally enter one or more headers.

Name	Value
Enter header name	Enter header value

Optionally enter the web request body.

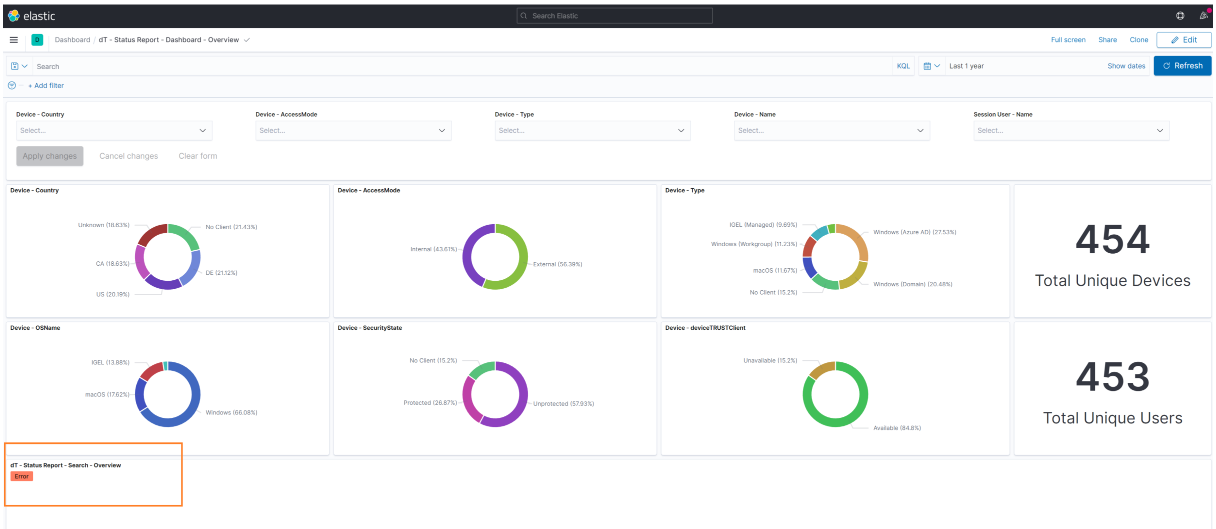
```
{
   "Session Date": "%TRIGGER_TIME%",
   "Device Name": "%HOST_REMOTECONTROL_REMOTE_NAME%",
   "Access Mode": "%CONTEXT_ACCESS_MODE%",
   "Anti-Virus Name": "%CONTEXT_ANTI_VIRUS_NAME%",
   "Anti-Virus Status": "%CONTEXT_ANTI_VIRUS_STATUS%",
   "Anti-Virus Timestamp": "%CONTEXT_ANTI_VIRUS_TIMESTAMP%",
   "Country": "%CONTEXT_COUNTRY%",
```

□ Wait for the web request to complete before continuing

□ Send the web request even when the content is unchanged

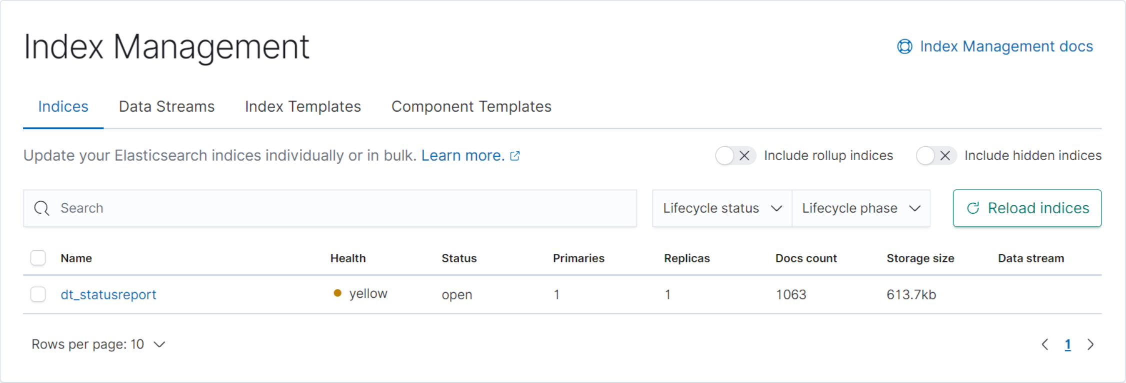
Step 3.4: Edit index settings

For the Status Report Dashboards to work properly, a configuration needs to be made at the index level: In a basic setting, ELK Stack allows to use 25 "calculated fields"per index. For the Status Report Dashboard, 48 calculated fields are used. Thus, the `allowed number of calculated fields` needs to be set to a higher value.



You need to send data to ELK Stack first. Sending data will create the index with basic settings. It can then be edited.

- After sending your first data, you will find the index `dt_statusreport` has been created in the Index Management Menu.

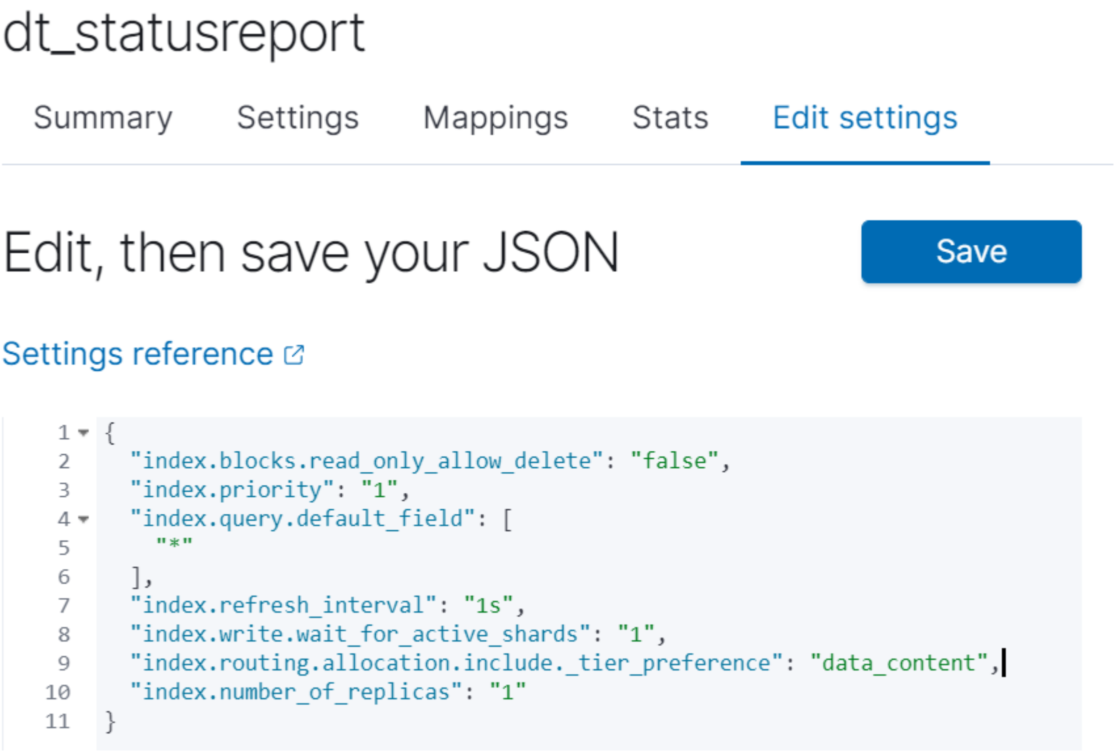


Index Management

Update your Elasticsearch indices individually or in bulk. Learn more.

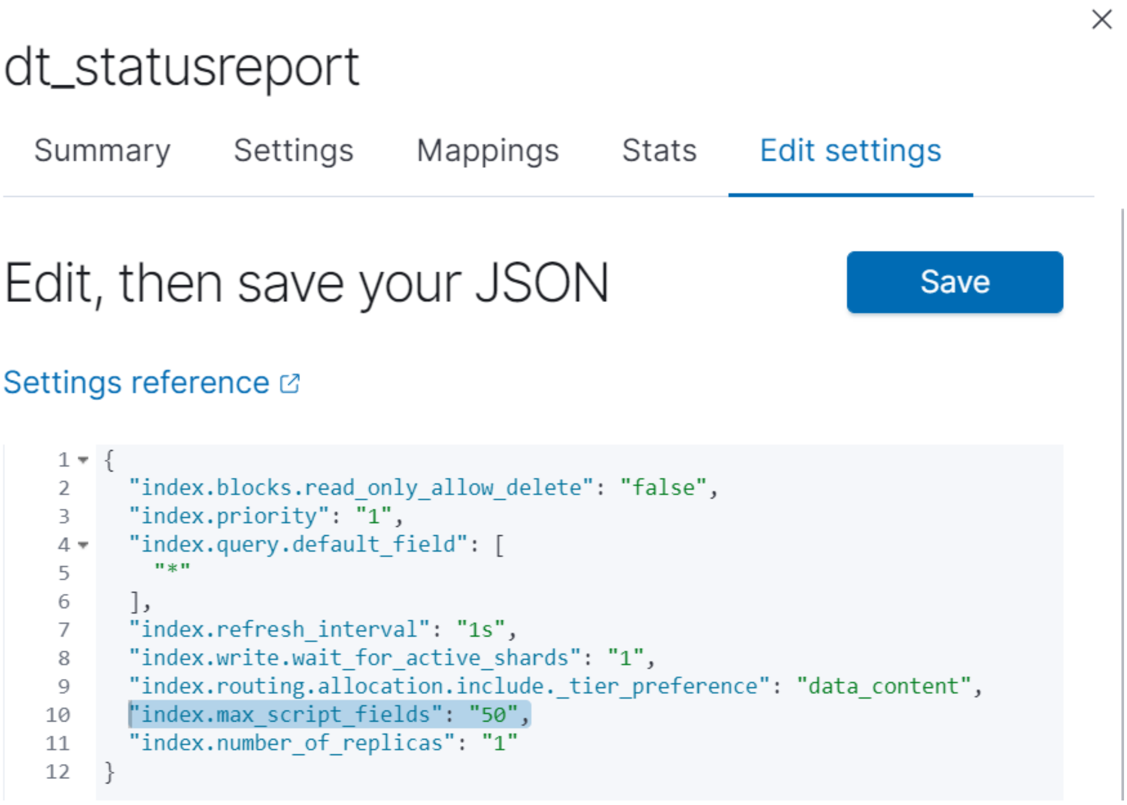
□	Name	Health	Status	Primaries	Replicas	Docs count	Storage size	Data stream
□	dt_statusreport	● yellow	open	1	1	1063	613.7kb	

- Select the Index and chose `Edit Settings`. You'll be presented a json configuration view.



```
{
  "index.blocks.read_only_allow_delete": "false",
  "index.priority": "1",
  "index.query.default_field": [
    "*"
  ],
  "index.refresh_interval": "1s",
  "index.write.wait_for_active_shards": "1",
  "index.routing.allocation.include._tier_preference": "data_content",
  "index.number_of_replicas": "1"
}
```

- Add `"index.max_script_fields": "50"` as a new line, making sure to keep the correct json formatting.
- Save your changes.



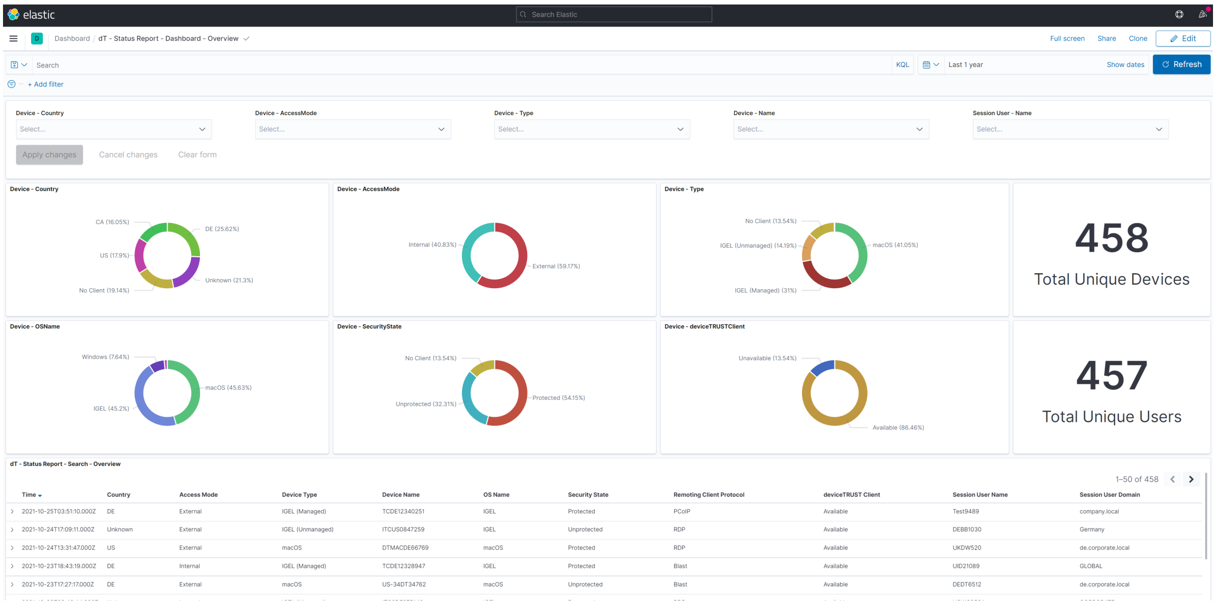
Edit, then save your JSON

```
{
  "index.blocks.read_only_allow_delete": "false",
  "index.priority": "1",
  "index.query.default_field": [
    "*"
  ],
  "index.refresh_interval": "1s",
  "index.write.wait_for_active_shards": "1",
  "index.routing.allocation.include._tier_preference": "data_content",
  "index.max_script_fields": "50",
  "index.number_of_replicas": "1"
}
```

Your Dashboard will now be displayed without errors.

After the index template has been created and the saved objects are imported, the use case `Status Report` has been implemented successfully.

deviceTRUST now sends status data to ELK Stack on every access to the remoting system. The data is presented in the created dashboards.



dT - Status Report - Search - Overview

Time	Country	Access Mode	Device Type	Device Name	OS Name	Security State	Remoting Client Protocol	deviceTRUST Client	Session User Name	Session User Domain
2021-10-25T03:51:10.000Z	DE	External	IGEL (Managed)	TCDE12340251	IGEL	Protected	PCoIP	Available	Test9489	company.local
2021-10-24T17:09:11.000Z	Unknown	External	IGEL (Unmanaged)	ITCUS0847259	IGEL	Unprotected	RDP	Available	DEBB1030	Germany
2021-10-24T13:31:47.000Z	US	External	macOS	DTMACDE66769	macOS	Protected	RDP	Available	UKDW520	de.corporate.local
2021-10-23T18:43:19.000Z	DE	Internal	IGEL (Managed)	TCDE12328947	IGEL	Protected	Blast	Available	UID21089	GLOBAL
2021-10-23T17:27:17.000Z	DE	External	macOS	US-34DT34762	macOS	Unprotected	Blast	Available	DEDT6512	de.corporate.local

Reference

TABLE OF CONTENTS

Properties

- Access Point Properties - Describes the available Wi-Fi access points.
- Browser Properties - Describes installed internet browsers.
- Cellular Properties - Describes the active cellular capabilities of an endpoint.
- Certificate Properties - Describes the private certificates available within the users certificate store.
- ChromeOS Properties - Provides properties unique to a Google ChromeOS device.
- Custom Properties - Provides a condition that can operate on any property, including custom properties created using 'dtcmd.exe'.
- deviceTRUST Properties - Provides information about whether a connection to a deviceTRUST Client Extension has been established, the version of the deviceTRUST software and the status of the deviceTRUST license.
- Display Properties - Describes the displays available to the user session.
- Domain Properties - Describes the domain membership of the endpoint.
- eLux Properties - Provides properties unique to a Unicon eLux device.
- Hardware Properties - Describes the hardware and its capabilities.
- IGEL Properties - Provides properties unique to an IGEL device.
- Input Properties - Describes the input devices available to the user session.
- iOS Properties - Provides properties unique to an Apple iOS endpoint.
- Location Properties - Describes the geographical location of the endpoint.
- Logical Disk Properties - Describes the logical disks available to the user.
- macOS Properties - Describes properties unique to an Apple macOS endpoint.
- macOS Firewall Properties - Provides real-time properties describing the state of the macOS Firewall.
- macOS Update Properties - Describes the status of macOS Software Update settings and updates.
- Mapped Drive Properties - Describes the mapped drives available within a user session.
- MDM Properties - Provides dynamic properties describing the current mobile device management (MDM) solution.

Agent Reference

TABLE OF CONTENTS

Client Extension Reference

TABLE OF CONTENTS

Console Reference

TABLE OF CONTENTS

Troubleshooting

If your deviceTRUST installation or configuration does not work as expected, then the Knowledge Base is a useful resource for common problems and resolutions. The following sections detail some useful knowledge base articles depending upon your deployment scenario:

Scenario: Remote

In remote scenarios, deviceTRUST® transports the context information from the user's remote device to the virtual session where the configuration is enforced. Please check the following knowledge base articles:

- Step 1: Make sure that you have a valid license
- Step 2: Check that your contextual security policy has been saved and deployed
- Step 3: Check that the user is managed by deviceTRUST
- Step 4: Check that the deviceTRUST Client Extension is installed on the remote device
- Step 5: Check that Citrix Virtual Channel Security is configured (Citrix Only)
- Step 6: Check that your contexts are correctly defined
- Step 7: Exclude specific users from the deviceTRUST policy
- Step 8: Check that the deviceTRUST Agent service is running
- Step 9: Check that you are using the latest deviceTRUST version

Scenario: Local

In local scenarios, deviceTRUST collects context information and executes actions locally. Please check the following knowledge base articles:

- Step 1: Make sure that you have a valid license
- Step 2: Check that your contextual security policy has been saved and deployed
- Step 3: Check that the user is managed by deviceTRUST
- Step 4: Check that your contexts are correctly defined
- Step 5: Exclude specific users from the deviceTRUST policy
- Step 6: Check that the deviceTRUST Agent service is running
- Step 7: Check that you are using the latest deviceTRUST version

Open a support ticket with us

Additional articles may be found by searching the Knowledge Base. However, if you are still experiencing difficulties please raise a ticket with the Citrix Support Portal at https://support.citrix.com.

Knowledge Base

TABLE OF CONTENTS

General

TABLE OF CONTENTS

Configuration

Table of Contents

Connectivity

Table of Contents

Diagnostics

Table of Contents

Installation

TABLE OF CONTENTS

Licensing

TABLE OF CONTENTS

Support

TABLE OF CONTENTS

Compatibility

Table of Contents

Properties

TABLE OF CONTENTS

Reporting

TABLE OF CONTENTS

Features

Table of Contents

Releases

TABLE OF CONTENTS

TABLE OF CONTENTS

June 20, 2025

IGEL OS 11 Client Extension 23.1.400

This release includes minor enhancements to the deviceTRUST Client Extension for IGEL OS 11. This release has been submitted to IGEL for native integration into IGEL OS 11, however this has not been released yet. To integrate into currently available IGEL OS releases, please consult our Updating the deviceTRUST Client Extension in IGEL OS 11 Knowledge Base article.

Support for third party WHOIS providers in 23.1.300

The IGEL OS 11 Client Extension 23.1.400 includes compatibility with the WHOIS changes introduced in deviceTRUST 23.1.300, including support for the IP2Location Web Services WHOIS Provider and the MaxMind GeoIP and GeoLite Web Services WHOIS Provider.

Compatibility

There are no other compatibility concerns with this release of the deviceTRUST® IGEL OS 11 Client Extension.

IGEL OS 12 Client Extension 23.1.400

We are very proud to make available our first deviceTRUST Client Extension for IGEL OS 12 devices. This release is available to install from the IGEL App Portal.

Release 23.1.200 includes bugfixes and minor enhancements to the deviceTRUST Client Extension.

Release 23.1.210 includes bugfixes to the deviceTRUST Client Extension to ensure compatibility with IGEL OS 12.3.1 and later.

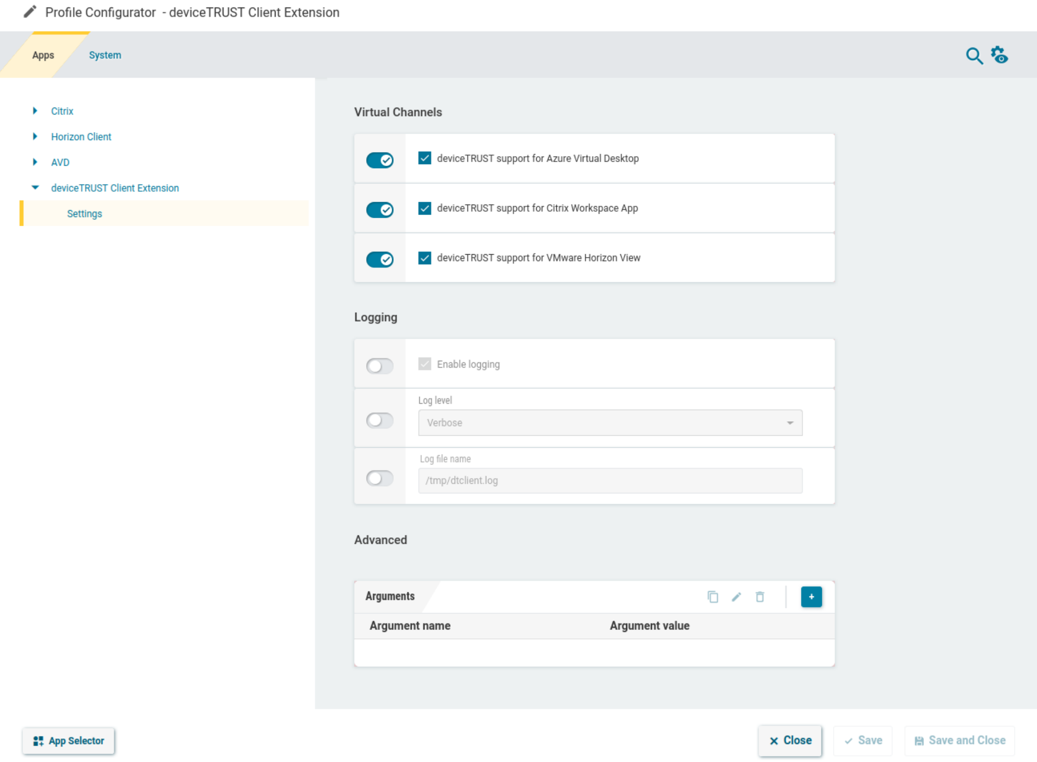
Release 23.1.400 adds support for WHOIS proxy, proxy type and usage introduced in release 23.1.400 of the deviceTRUST Agent and Console.

Remoting Clients

IGEL OS 12.01.120 and later supports native integration of the deviceTRUST Client Extension for Citrix Workspace App, Microsoft Azure Virtual Desktop and VMware Horizon View. For more information, see Supported IGEL Operating Systems for Client Extension.

UMS Configuration

The deviceTRUST® Client Extension for IGEL OS 12 can be configured directly within the UMS 12 Web App.


Profile Configurator - deviceTRUST Client Extension
Apps
System
Citrix
Horizon Client
AVD
deviceTRUST Client Extension
Settings
App Selector
Close
Save
Save and Close

Virtual Channels

- ☑ deviceTRUST support for Azure Virtual Desktop
- ☑ deviceTRUST support for Citrix Workspace App
- ☑ deviceTRUST support for VMware Horizon View

Logging

- ☑ Enable logging
- Log level: Verbose
- Log file name: /tmp/dtclient.log

Advanced

Arguments

Argument name	Argument value

New COSMOS Properties

IGEL OS 12 devices no longer distinguish between a UMS and ICG server. As a result, the previous ICG and UMS properties are no longer populated, and the following properties have been added to the IGEL category:

- **COSMOS Server** - The name and port of the COSMOS server.
- **COSMOS Cert Count** - The number of certificates from the COSMOS chain of trust.
- **COSMOS Cert Serial** - The serial number of the certificate from the COSMOS chain of trust.
- **COSMOS Cert Subject** - The subject of the certificate from the COSMOS chain of trust.
- **COSMOS Cert Thumbprint SHA256** - The SHA256 thumbprint of the certificate from the COSMOS chain of trust.

More information can be found at IGEL Properties.

WHOIS proxy, proxy type and usage in 23.1.400

Support for the detection of WHOIS proxy, proxy type and usage via either IP2Location Web Services or MaxMind GeoIP and GeoLite Web Services has been added in release 23.1.400.

Bug fixes in 23.1.200

- Fixed an issue where COSMOS Server property was empty even when the device was managed.

Minor enhancements in 23.1.200

- Added support for the Volume ID of Logical Disk Properties on Linux based devices.
- Network Properties are now dynamic on Linux based devices.
- Logging of the deviceTRUST Client Extension is now disabled by default. Logging can be enabled within the UMS Configuration.

Bug fixes in 23.1.210

- Fixed an issue where some IGEL COSMOS properties could not be queried on IGEL OS 12.3.1 or later.
- Fixed an issue where the Hardware Bios Serial property appeared as Unavailable.
- Fixed an issue with VMware Horizon View where the deviceTRUST Client Extension would not load when using IGEL OS 12.3.1 or later.

Compatibility

There are no other compatibility concerns with this release of the deviceTRUST IGEL Client Extension.

iOS Client Extension 23.1.400

This release brings support for iOS and iPadOS 17 and later, and is available to download from the App Store.

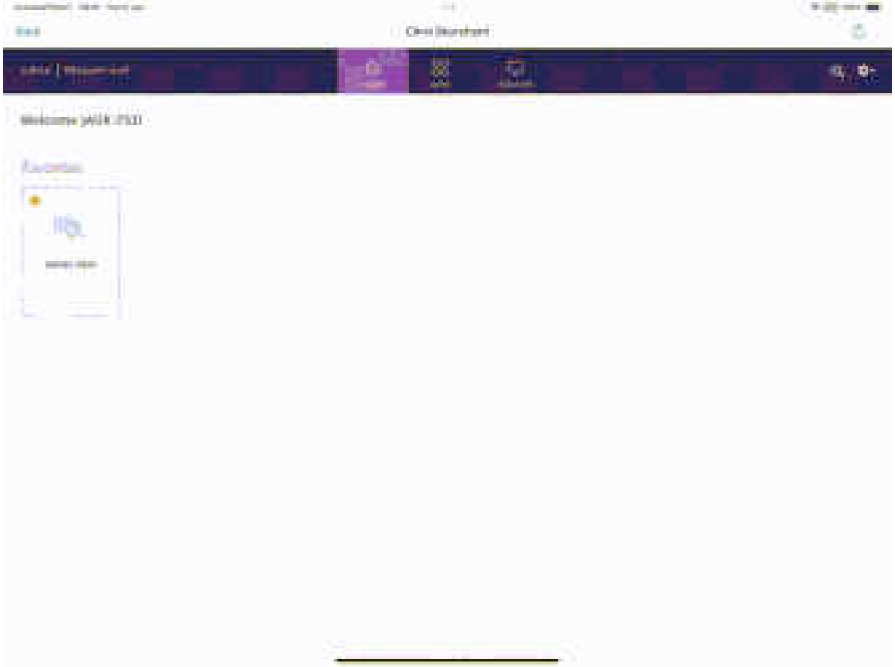
Release 23.1.400 includes support for WHOIS proxy, proxy type and usage. This functionality required version 23.1.400 of the deviceTRUST Agent and Console.

Please refer to the Client Extension installation on iOS and iPadOS devices for more information.

Passcode authentication for iOS 17

Previously when connecting to a virtual session with an iOS device, the deviceTRUST Agent matched the *client name* taken from the remoting protocol with the name of the registered iPhone or iPad within the deviceTRUST Portal. However with Apple's continued push towards user privacy, the ability to query the name of the device is now limited to apps meeting specific criteria and requesting the necessary entitlement. Neither the deviceTRUST Client Extension for iOS, or any of the popular remoting clients, have been granted this entitlement. Having lost the only information we have that identifies the remote iOS device, a new approach to matching the remote device is required.

We now support the use of passcodes to match the remote device. When connecting from a remote iOS device, the deviceTRUST® Agent will now display a passcode to the user of configurable length. The user must then switch to the deviceTRUST Client Extension for iOS and enter that passcode. They can then return to the virtual session to gain access to their desktop or published application.

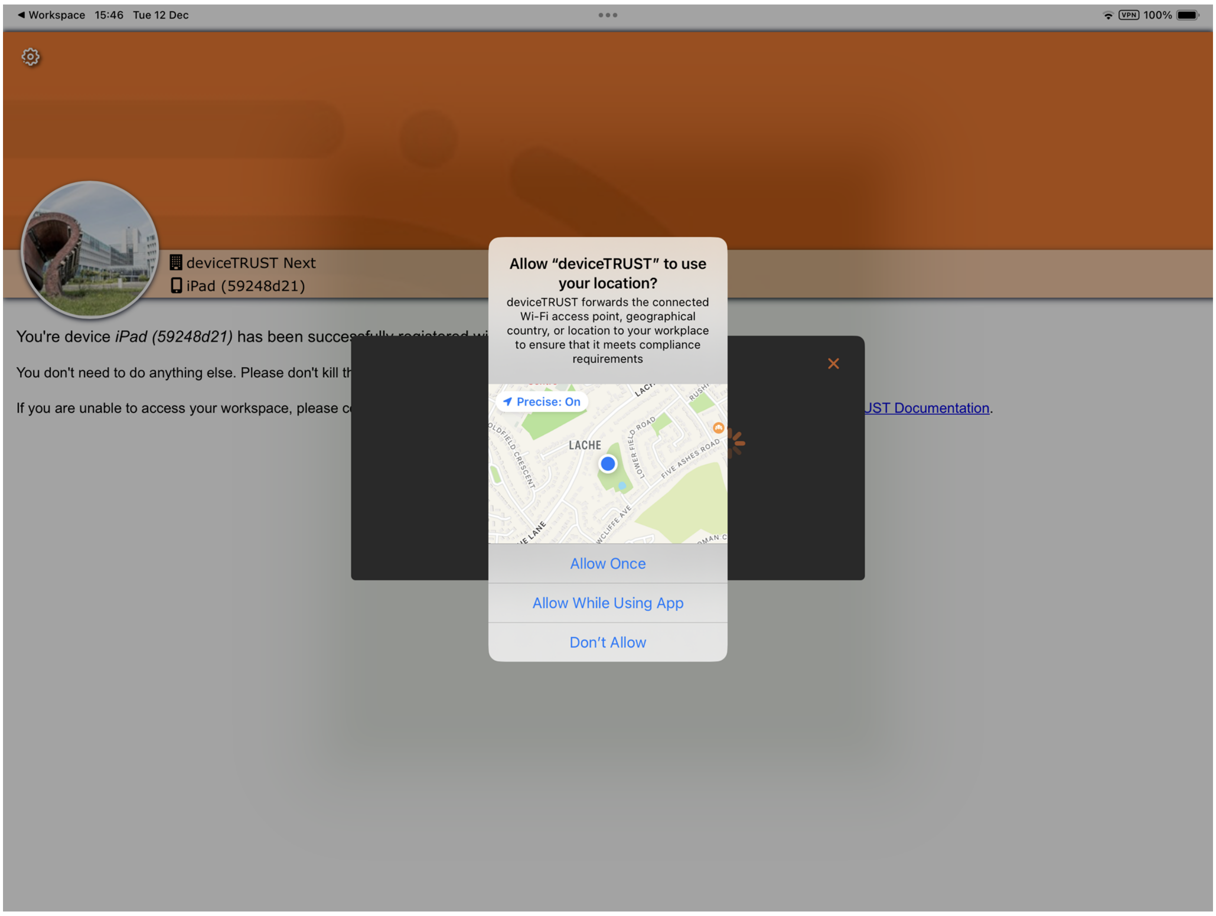


By using passcodes, we now support iOS as a remote device on Citrix Virtual Apps and Desktops, Citrix Cloud, Microsoft Remote Desktop Services and VMware Horizon View. For more information, see OS Compatibility.

More information can be found in the iOS Passcodes reference.

Location and Network Wi-Fi SSID/BSSID

We've added support for Location and Network Wi-Fi SSID and BSSID properties from iOS remote devices. Users must consent to location privileges when either Location or Network Wi-Fi SSID and BSSID are first requested. Since these properties potentially require user interaction, they are only available when using passcode authentication.



deviceTRUST Next

iPad (59248d21)

You're device *iPad (59248d21)* has been succe

You don't need to do anything else. Please don't kill th

If you are unable to access your workspace, please c

UST Documentation.

Allow "deviceTRUST" to use your location?

deviceTRUST forwards the connected Wi-Fi access point, geographical country, or location to your workplace to ensure that it meets compliance requirements

Compatibility

This version of the deviceTRUST Client Extension for iOS is backwards compatible with previous deviceTRUST Agents. However the new passcode authentication functionality is required for compatibility with iOS 17 and later, and requires deviceTRUST Agent 23.1.200 or later.

macOS Client Extension 23.1.410

This release includes enhancements and bug fixes to the deviceTRUST® Client Extension for macOS.

Release 23.1.110 includes a bug fix to address changes in macOS Sonoma that now require apps to request location permission to be able to access Wi-Fi information of the connected networks.

Release 23.1.200 includes a bug fix to address compatibility issues with macOS 10.15 Catalina.

Release 23.1.300 includes compatibility with third party WHOIS providers.

Release 23.1.400 includes bug fixes to address compatibility issues with macOS 15.0 Sequoia, as well as support for new Microsoft Intune and WHOIS properties introduced in release 23.1.400 of the deviceTRUST Agent and Console.

Release 23.1.410 includes a single bug fix to the security properties.

Support for third party WHOIS providers in 23.1.300

The macOS Client Extension 23.1.300 includes compatibility with the WHOIS changes introduced in deviceTRUST 23.1.300, including support for the IP2Location Web Services WHOIS Provider and the MaxMind GeoIP and GeoLite Web Services WHOIS Provider.

Support for Microsoft Intune detection and WHOIS proxy, proxy type and usage in 23.1.400

Support for the detection of Microsoft Intune, and for WHOIS proxy, proxy type and usage via either IP2Location Web Services or MaxMind GeoIP and GeoLite Web Services has been added in release 23.1.400.

Nice DCV extension support including Amazon WorkSpaces WSP

We now include a Nice DCV extensions within the deviceTRUST Client Extension for macOS bringing support for the latest Nice DCV Standalone, Amazon WorkSpaces WSP, Amazon AppStream and more.

On Amazon WorkSpaces WSP, the `Configure extensions` policy must be set to `Server and Client` and there are additional software requirements. More information can be found in our Enabling DCV extensions on Amazon WorkSpaces WSP knowledge base article.

Bug Fixes in 23.1.100

- Fixed an issue where the macOS Update Enabled property could be inaccurate on macOS 13.0 Ventura and later.

Bug Fixes in 23.1.110

- Fixed an issue where the Network Wi-Fi properties were inaccurate on macOS 14 .0 Sonoma, with the *Wi-Fi Security* property was set to “Other (-1)”and *Wi-Fi BSSID* and *Wi-Fi SSID* both empty. This issue was caused by macOS Sonoma now requiring permission to access the user location in order to determine these properties, and as a result the deviceTRUST Client Extension for macOS now prompts for location permissions when these properties are requested.

Bug Fixes in 23.1.200

- Fixed an issue where the deviceTRUST Client Extension would fail to load on macOS 10.15 Catalina.

Bug Fixes in 23.1.400

- Fixed an issue with macOS 15.0 Sequoia where the macOS Firewall properties were all Unavailable. As part of these changes, the Version, Inbound Exception Rules and Inbound Service Rules properties have been deprecated.
- Fixed an issue with macOS 15.0 Sequoia where the Location properties were initially set to Unavailable. The location properties provided by macOS 15.0 Sonoma can take longer to load than in previous OS releases, so support for the Loading state has been added.

Bug Fixes in 23.1.410

- Fixed an issue in the 23.1.400 release where the macOS Firewall must have Stealth Mode enabled for the firewall to appear as Active within the Security Products properties.

Known Issues

- Amazon WorkSpaces no longer allows PCoIP virtual channels to be loaded. This can be worked around by installing Amazon WorkSpaces Client v5.3.0.

Compatibility

If you are using the macOS Firewall's Version, Inbound Exception Rules or Inbound Service Rules properties within a Context, these conditions should be removed.

For Amazon WorkSpaces WSP support, please ensure you are using deviceTRUST 23.1.112 or later to ensure reliable connectivity between the deviceTRUST Agent and the deviceTRUST Client Extension for macOS.

Stratodesk NoTouch OS Client Extension 23.1.100

We are very proud to make available our first deviceTRUST Client Extension for Stratodesk NoTouch devices. This first release includes support for Citrix Workspace App, FreeRDP and VMware Horizon View. Please check out Client installation on Stratodesk NoTouch OS devices for details on the installation. This client extension requires deviceTRUST Agent and Console version 23.1.100 or later.

Available Properties

We've added support for the following categories of properties:

- deviceTRUST - Provides the version of the deviceTRUST software.
- Display - Describes the displays available to the user session.
- Hardware - Describes the hardware and its capabilities.
- Input* - Describes the input devices available to the user session.
- Location - Describes the geographical location of the endpoint.
- Logical Disk - Describes the logical disks available to the user.
- Name - Identifies the endpoint.
- Network - Describes the network adapters and their bound network addresses.
- NoTouch - Provides properties unique to a Stratodesk NoTouch device.
- OS - Provides information about the operating system installed on the endpoint.
- [Power]/en-us/device-trust/reference/properties/power/) - Describes the power profile of the endpoint.
- Region - Describes the regional information of the user session.
- Smartcard Reader - Describes the connected smart-card readers available to the user.
- User - Identifies the logged in user.
- WHOIS - Provides the results of a WHOIS lookup of the endpoint.

NoTouch Properties

The following properties are available within the NoTouch category:

- **NoTouch Center URL** - The Management URL of the NoTouch Center.
- **NoTouch Center Cert Serial** - The serial number of the certificate used to connect to the NoTouch Center.
- **NoTouch Center Cert Subject** - The subject of the certificate used to connect to the NoTouch Center.
- **NoTouch Center Cert Thumbprint SHA256** - The SHA256 thumbprint of the certificate used to connect to the NoTouch Center.
- **Environment Variable Count** - The number of NoTouch Center environment variables.
- **Environment Variable Name** - The name of the NoTouch Center environment variable.
- **Environment Variable Value** - The value of the NoTouch Center environment variable.

Custom Properties for NoTouch Devices

The Custom Properties Settings includes support for remote NoTouch OS devices. This allows a custom script to return any property from the remote device by writing `REMOTE_CUSTOM_NAME=VALUE` to the output.

Ubuntu Client Extension 23.1.400

September 6, 2025

Release 23.1.300 includes support for third party WHOIS providers.

Release 23.1.400 includes support for WHOIS proxy, proxy type and usage.

Support for third party WHOIS providers in 23.1.300

The Ubuntu Client Extension 23.1.300 includes compatibility with the WHOIS changes introduced in deviceTRUST 23.1.300, including support for the IP2Location Web Services WHOIS Provider and the MaxMind GeoIP and GeoLite Web Services WHOIS Provider.

Support for WHOIS proxy, proxy type and usage in 23.1.400

Support for the detection of WHOIS proxy, proxy type and usage via either IP2Location Web Services or MaxMind GeoIP and GeoLite Web Services has been added in release 23.1.400.

Bug Fixes in 23.1.300

- Fixed an issue loading the virtual channels on Ubuntu 22.04 or later due to a missing dependency on libssl.

Bug Fixes in 23.1.400

- Fixed an issue where wireguard VPN devices would not identify a VPN within the WHOIS properties.

Compatibility

There are no compatibility concerns with this release of the deviceTRUST® Ubuntu Client.

deviceTRUST 23.1.410

This service pack adds detection for Microsoft Intune and introduces the detection of proxies using IP2Location Web Services or MaxMind GeoIP2 Insights Web Service providers. Please refer to Compatibility for changes that may impact users upgrading from previous releases.

The deviceTRUST 23.1.410 patch includes bugfixes to the deviceTRUST Agent and Console. This release includes no changes to the deviceTRUST Client Extension and therefore has not been released on https://devicetrust.com/download.

Microsoft Intune Detection

Microsoft Intune can now be detected using our new MDM properties. Microsoft Intune detection is supported on Microsoft Windows and Apple macOS devices.

WHOIS Proxy Detection

The WHOIS properties have been extended with support for proxy, proxy type and usage properties. These properties are taken from the response fields of the IP2Location or MaxMind provider when an appropriate plan is selected.

As part of this change, we've renamed `LOCAL_WHOIS_VPN` and `REMOTE_WHOIS_VPN` to `LOCAL_WHOIS_ADAPTER_VPN` and `REMOTE_WHOIS_ADAPTER_VPN` respectively to ensure clarity with the new proxy properties.

Proxy Detection with IP2Location Web Services

IP2Location Web Services provides proxy detection from the following IP2Location response fields and is available with the following plans:

Property	Response Field	Free Plan	Starter Plan	Plus Plan	Security Plan
Proxy	`is_proxy`<sup>1</sup>	✓	✓	✓	✓
Proxy Type	`proxy.proxy_type`				✓
Usage	`usage_type`		✓	✓	✓

> **Note:**
>
> The accuracy of the `is_proxy` field is dependent upon the selected plan.

Proxy Detection with MaxMind GeoIP2 Insights Web Service

MaxMind GeoIP and GeoLite Web Services provides proxy detection from the following MaxMind response fields and is available with the following plans:

Property	Response Field	GeoIP2 Country	GeoIP2 City Plus	GeoIP2 Insights	GeoLite Country	Geolite City
Proxy	`is_anonymous`			✓		

Proxy Type	`is_hosting_provider`
	,
	`is_anonymous_vpn`
	,
	`is_public_proxy`
	,
	`is_residential_proxy`
	,
	`is_tor_exit_node`
				✓		

Usage	`user_type`			✓		

Minor enhancements to 23.1.400

- Renamed Microsoft Azure AD to Microsoft Entra ID throughout the console.
- Added a `Use remote client names to connect to iOS devices` advanced setting, which forces the deviceTRUST® Agent to connect to iOS devices by matching the remote client names with those registered within the iOS Portal.

Bug fixes in 23.1.400

- Fixed an issue where a virtual session could be logged off during a live reconnect to a session that had never shown the desktop to the user.
- Fixed an issue with the Logical Disk Identity operator where the None Of operator was shown as Not Equals.

Bug fixes in 23.1.410

- Fixed an issue on Citrix® virtual sessions when running a client OS, where the logoff process could get stuck and require a restart of the deviceTRUST Agent before allowing additional logons.
- Fixed an issue within the deviceTRUST Console where a context custom condition would fail to load when the custom condition targeted data types of IP or Position.

Compatibility

This compatibility section builds on our general approach to compatibility which can be found on the compatibility page.

We've renamed `LOCAL_WHOIS_VPN` and `REMOTE_WHOIS_VPN` to `LOCAL_WHOIS_ADAPTER_VPN` and `REMOTE_WHOIS_ADAPTER_VPN` respectively to ensure clarity with the new proxy properties. deviceTRUST Policies will upgrade automatically, but if these properties are referenced within scripts, or Web Request tasks, they may need to be manually updated.

If upgrading from a release prior to 23.1.400, be sure to check out the deviceTRUST 23.1.300 compatibility notes.

The deviceTRUST Agents can read policies created by previous releases of the deviceTRUST Console. However, they cannot read policies created by a newer console. Therefore, you must ensure that the deviceTRUST Agent 23.1.400 is deployed before applying policy that has been written by the deviceTRUST Console 23.1.400 or later.

June 20, 2025