7

# Managing and Maintaining the Security Posture

*Good preparation is half the battle won* and *good tools are half the work* are two maxims that come to mind at this point in our journey. Just beyond midway, with tools and preparation in hand, we are now going to see how to sustainably run MDE in your environment and set yourself up for continued success. However, here comes the hard part – **continuous security posture management**. This is where perseverance, grit, and grinding to stay ahead of the eternal cat-and-mouse game come into play. If you can nail this part, you will make life a lot easier for everyone else.

When it comes to the maintenance of device health and ensuring a security stack is healthy and current, you could argue that it is a pivotal responsibility for you and your team. You have put in so much work up to this point, many hours of reading and testing so that things go as well as they can during planning, deployment, and configuration, so why stop there? This should be a continual upkeep of your investment.

We're going to unpack four key areas to get a deeper sense of what is involved with each:

- Performing production readiness checks
- Staying up to date
- Maintaining your security posture through continuous discovery and health monitoring
- Getting started with vulnerability management

## Performing production readiness checks

For production deployment, here are some of the things you may want to check before proceeding with broad deployment in your environment.

### Considerations for connectivity

MDE is driven by cloud services. Making sure your endpoints can reach cloud service endpoints is one of the most important things to do. If you haven't already, the MDE client analyzer tool (**https://aka.ms/mdeanalyzer**) will help you check whether everything is in working order before proceeding to ***Chapter 9***, *Troubleshooting Common Issues*.

If you are using a proxy, here are some additional considerations:

- A system-wide proxy works best, meaning that for any HTTP or HTTPS request, the operating system can try to connect through the proxy, including MDE cloud services. You will then allow those connections in the proxy itself if you wish to pursue an allow-list approach (typically appropriate for servers but not user endpoints). A system-wide proxy can be configured in a variety of ways, including automatically, coming from a network (using **Web Proxy Auto-Discovery** (**WPAD**), for instance), using a transparent proxy, using a *static* configuration of `winhttp` for Windows, `http_proxy`/`https_proxy` on Linux, or using a user interface-based configuration (**Network preferences**) on macOS.
- If you only wish to use a proxy specifically to allow the MDE services on a device to connect to MDE cloud services, you can. This requires additional configuration through group policy or registry for Windows. Note that you need to configure three settings, two for telemetry/**Endpoint Detection and Response** (**EDR**) (`TelemetryProxyServer` and `DisableEnterpriseAuthProxy`) and one for Defender Antivirus (`ProxyServer`). On Linux, you can configure the proxy with `mdatp config proxy set` or by adding the setting to the `mdatp.service` configuration file.
- If a device cannot *freely* reach the broad internet, it likely also cannot reach Microsoft Update or the update package locations. This can impact a device's ability to receive regular updates for the app/platform and definitions. You will need to consider a patch management strategy or still allow certain updates to flow directly from online sources. Definition updates are an area where you particularly need to find a balance between control and automation, as they can arrive very frequently, and waiting for a sync to happen in your patch management system can cause more harm than good.
- As an additional security measure, MDE asks you to ensure your local **certificate trust list** is up to date to help validate the **chain of trust**, if the device cannot reach the online locations to verify the certificate chain.

## Enabling Defender Antivirus capabilities

For **Microsoft Defender Antivirus**, the main things you will want to consider are the following:

- Do you have your baseline policies defined, tested, and ready to go?
- Did you enable/disable the necessary capabilities for your organization?
- Do you have a way to place exclusions as needed?

There are baselines available to make it easier for you to select/enforce secure defaults. In general, you will want to make sure the following capabilities are on by default before making any exceptions:

- **Real-time protection**
- **Cloud-delivered protection**
- **Automatic sample submission**
- **Block at first sight**
- **Potentially unwanted application (PUA) blocking**

- **IOAV protection (scan all downloaded files and attachments)**
- **Tamper protection**

The last setting in the list, **Tamper protection**, enforces most of the preceding and even prevents a local administrator from overriding the secure defaults that were set on initial installation, as defined by Microsoft. During your pilot deployment, you may want this set to off – but afterward, it's one of the best things you can enable to protect your environment against attackers. This requires you to be on point to understand how to mitigate the potential impact of application compatibility issues or false positives. The **Troubleshooting mode** feature can help a lot here by allowing you to temporarily modify the configuration.

When it comes to scheduling scans, please consider that cloud-delivered protection provides a strong first line of defense that can protect against malware *on the fly*, and that you may not need to run full scans unless you have just onboarded a device and want to start *clean* – and build up the cache. You can read more about these considerations in ***Chapter 2***, *Exploring Next-Generation Protection*.

## Attack surface reduction

If you recall from ***Chapter 3***, *Introduction to Attack Surface Reduction*, we have three main categories of features we're looking to get rolled out. We have the **Attack Surface Reduction** (**ASR**) rules, **Controlled Folder Access** (**CFA**), as well as **Network Protection** (**NP**). In the next few sections, we'll talk about each one and what we should be considering for each as we go.

### ASR rules

A quick note before we get started here – to give **ASR** rules the ability to do anything, (and by anything, we mean audit, warn, or block) we need **Defender Antivirus** to be the active, primary antivirus.

Alright, you're rolling out ASR rules in production, so what should that look like? The way Microsoft approaches this guidance is by breaking rules into two categories, **standard protection rules** and **other rules**. What this essentially means is that the rules that fall into the standard protection bucket can be enabled without hesitation, and the others, because of the vast variation in environments across the globe, need some level of consideration before putting them into block mode. Of course, you should be considering which rules apply to what OS as well to avoid a good old rabbit hole, trying to figure out why something is not applying to a device.

Your main deployment options here are going to be through group policy, **Microsoft Intune**, and Microsoft **Configuration Manager** (**ConfigMgr**). PowerShell is an option too but not the ideal method for production rollouts – it will only set preferences (see ***Chapter 6***, *Considerations for Deployment and Configuration*, for more information about management tool selection). Note that if you're using ConfigMgr to deploy ASR rules, you will not see the **Block process creations originating from PSExec**

**and WMI commands** rule, and that is on purpose. If you are deploying rules via Intune or a **Group Policy Object** (**GPO**) and you are using ConfigMgr as well, please still refrain from using that rule. It will break your clients, as the ConfigMgr agent uses WMI commands to control a device.

Let's look at the rules in the standard protection bucket first, as those should be the ones enabled right away:

- **Block abuse of exploited vulnerable signed drivers**
- **Block credential stealing from the Windows local security authority subsystem (lsass.exe)**
- **Block persistence through WMI event subscription**

While there are likely many other rules you can enable in **block** mode right away, it's always best to start with **audit** mode and let them saturate so that you can see quality data and decide. Thirty days is the ideal amount of time in **audit** mode; this ensures you get a concrete look at your environment and allows for consistent use.

What does *saturate* mean? It's where you let a setting or configuration sit in your environment, on a representative sample of devices, long enough to be as certain as possible that it's had time to interact with all significant variations of the activity your organization's systems generate. There is a great blog out there written by Chad D on the https://blog.palantir.com/ site (`@duff22b` on Twitter) that goes over testing he did over an extended period, showing which rules are safe for most environments, ones that require a little more consideration, and ones that need a serious amount of focus before deploying in block mode. It provides great context into the detections, their volumes, as well as some guidance along the way. Check it out at *Microsoft Defender Attack Surface Reduction Recommendations*: https://tinyurl.com/ypysh3am.

Toward the end of this chapter, we'll cover some reporting options where you can check the status of the rules in place, the modes they're in, and what some of the detections coming in are, as well as some advanced hunting queries you can run to dig a little deeper into these detections. This will help you determine whether you need to get some exclusions in place before flipping to **block** mode.

### Controlled folder access

Let's move on to CFA; we will approach this one just like ASR rules and start with audit mode. Like most features that carry an audit mode, it's the place to start because it allows you to see what it does, see what it would have done, and what you need to do from an exclusion standpoint to be successful. That's no different here; you should be deploying this in audit mode and checking events to see what exclusions are needed.

CFA can be tricky when deploying. While it's safe to just deploy in block mode, be mindful that it can be noisy to an end user, generating lots of toast notifications to inform the user of blocks. This is due to the sheer number of apps trying to make changes to the user profile, which you will have likely already noticed by this point in the process.

Since CFA does not have any native reporting in the MDE portal, we'll share some items here that you can check after you have this out in production. The following query will give you a record of those events in the timeframe you select so that you can see how it's working:

```
DeviceEvents
| where ActionType in ('ControlledFolderAccessViolationAudited','ControlledFolderAccessViolation
```

*COLD SNACK*

*CFA events do not trigger alerts in MDE; they are simply logged and sent to the device timeline as an event. If you want to alert, consider creating custom detection rules.*

Once you have dialed in CFA, a great next step is to go and create custom detection rules to generate related alerts on machines of interest, or in general across an environment.

### Network protection

Let's move on to NP; what does that look like? Before proceeding, we need to ensure a few things are in place when we enable and expect NP to do anything. We need **Real-Time Protection** (**RTP**) and **Cloud-Delivered Protection** (**CDP**) enabled, and so again we need Defender Antivirus to be the active, primary antivirus.

**Audit** mode is the way to go here as well. You should see a pattern by now, as discussed in ***Chapter 3***, *Introduction to Attack Surface Reduction*, and it's the theme here as well. Assuming you started that way and have deployed to production, you want to check how it's going. Like CFA, there are not really any dedicated reports for NP. On its own, it does not generate any alerts; the output is simply relayed to the device timeline as events that are obtainable via **advanced hunting**:

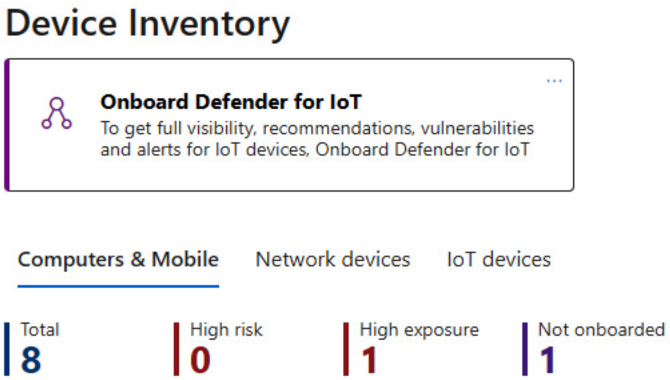
```
DeviceEvents
|where ActionType contains "ExploitGuardNetworkProtection"
|extend ParsedFields=parse_json(AdditionalFields)
|project DeviceName, ActionType, Timestamp, RemoteUrl, InitiatingProcessFileName, IsAudit=tostri
|sort by Timestamp desc
```

Once you start seeing data coming in and actions being audited, make your adjustments and start deciding on when you want to flip it to **block** mode.

## Endpoint detection and response

When it comes to EDR, your production deployment should be underway by now, and devices should be reporting in and beginning to deliver telemetry to the platform. One quick thing you can do to start spot-checking your deployments is to check the **Device Inventory** page in the portal. You'll see the counts going on, and likely some are displaying **Not onboarded** if some areas in your environment were not explicitly targeted.

If you start to notice a pattern of the devices showing up as **Can be onboarded**, it is likely that they all need the same resolution; perhaps they're behind some other network appliance. The following screenshot gives you an example of the breakdown in the header of that page:



Device Inventory

**Onboard Defender for IoT**
To get full visibility, recommendations, vulnerabilities and alerts for IoT devices, Onboard Defender for IoT

| Total | High risk | High exposure | Not onboarded |
|---|---|---|---|
| 8 | 0 | 1 | 1 |

Figure 7.1 – Device inventory breakdown

Outside of devices reporting in, there isn't much to do other than ensuring your device groups are sorted out and, more explicitly, setting the automation levels so that devices can start acting on alerts should something arise. Refer back to ***Chapter 4***, *Understanding Endpoint Detection and Response*, for recommendations on automation levels. (The short of it is that you should turn automation on everywhere unless you have a good reason not to.)

With the final checks completed, we will move to production. Of course, you will want to have your SOC team ready to operate – check out ***Chapter 8***, *Establishing Security Operations*, for more information on that. But now comes the hard part – keeping everything running smoothly and securely.

## Server-specific settings

Some settings have particular significance when it comes to server OSs. Here are some settings that need to be carefully considered on servers.

### Automatic exclusions

The **automatic exclusions** feature on **Windows Server 2016** and later can help make it easier to ensure little or no exclusions management is needed for standard Windows Server roles. However, the following items do require some attention:

- Automatic exclusions only apply if the roles were installed in the default location. Especially with domain controllers, this may not be the case.

- You may not want these automatic exclusions for your specific workload – remember, exclusions impact security as well. You can turn them off.
- Contextual exclusions may be better suited, as they allow you to be more specific.

The trade-off here is the ease of management versus security.

### Network protection and related settings

The overhead that comes with NP can, especially on servers that deal with a lot of network traffic or connections, impact performance significantly. In particular, **User Datagram Protocol** (**UDP**) processing is very impactful, which is why it comes with its own separate enablement commands:

- `Set-MpPreference -EnableNetworkProtection Enabled`
- `Set-MpPreference -AllowNetworkProtectionOnWinServer 1`
- `Set-MpPreference -AllowNetworkProtectionDownLevel 1`
- `Set-MpPreference -AllowDatagramProcessingOnWinServer 1`

For Linux machines, you will likely want to use audit mode to examine potential impact – however, note that even audit mode has performance implications!

The following command will set NP for Linux to audit mode:

```
sudo mdatp config network-protection enforcement-level --value audit
```

### Real-time scanning direction

You can configure RTP to only scan when files are either coming in or going out of a filesystem. Take file servers as an example. They have lots of outbound traffic by the nature of their job, and scanning the flow in that direction might not be valuable enough to warrant the performance hit. Therefore, you may want to make a conscious choice on which direction(s) should be enabled on busy machines.

The setting in GPO is called **Configure monitoring for incoming and outgoing file and program activity**.

For our file server example, where outbound reads are from clients that should be covered by their own real-time protection, inbound reads may be more useful to scan, as those would constitute new files, and it would be good to scan them before storing them.

Note that the performance benefit will typically be minimal under normal circumstances.

### Multi-session environments (remote desktop services)

When you are running a server OS to serve shared desktops, a few things are important to keep in mind:

- Run scheduled scans outside of working hours.
- Disable automatic scans after definition updates. If needed, you can reduce the frequency of definition updates if you ensure that CDP is enabled and working.
- Do not enable features that require user interaction. Some specific examples are *putting capabilities in warn mode* or *having a user make a decision around automatic remediation*.
- Disable the user interface for users.

### Passive mode

On Windows and Linux servers, Defender Antivirus does not automatically move into passive mode if you install a third-party antivirus (the only reason to run in passive mode!). The reason for this is that the Windows Security Center service, which is the arbiter for orchestrating multiple antimalware solutions on Windows 10 and later, is not present.

If you wish to run Defender Antivirus in passive mode or **EDR block** mode, you will need to explicitly configure it to do so using the `ForceDefenderPassiveMode` registry key. Note that this setting is protected by **Tamper protection**; it will allow you to toggle it off, meaning Defender Antivirus will go into active mode, but not on, to go into passive mode, after you have onboarded the machine.

For an in-depth explanation of passive mode and when it is applicable, refer to the *Running modes* section of ***Chapter 2***, *Exploring Next-Generation Protection*.

Now that you're ready to move forward with your production deployment, it's time to think ahead – how will you ensure you get the best security coverage by staying up to date?

## Staying up to date

The different components of MDE need to be kept up to date – and it's not just security intelligence updates you need to stay on top of. Here are the key component updates you need to be ready to deploy on a regular basis *after* you install MDE.

*COLD SNACK*

*Even if you are running Defender Antivirus in passive mode, you will want to update it regularly to ensure you have the latest capabilities and protection for features that depend on it.*

### Windows

As outlined in ***Chapter 2***, *Exploring Next-Generation Protection*, for Windows, Defender updates can flow from **Windows Update** or **Microsoft Update**. They can also come straight from the **Microsoft**

**Security Intelligence** page (**https://www.microsoft.com/en-us/wdsi**), either as a manual download or as a fallback location. You can define the order using configuration.

In *Table 7.1*, we see the breakdown of cadences for the various components, available update methods, and update mediums:

| Component | Method | Sources | Cadence |
|---|---|---|---|
| **Security Intelligence** | Standalone package (`mpam-fe.exe`). Contains both the full engine and the latest definitions | Manual download from the Microsoft Security Intelligence page | Used to get up to speed. The full package is also used automatically if the device is heavily outdated |
| | Definitions only | Windows Update: **definitionupdates.microsoft.com** | Daily deltas |
| | New engine | Windows Update: **definitionupdates.microsoft.com** | Monthly full package – deltas for definitions |
| **Antimalware platform** | Standalone package (`updateplatform.exe`) – KB4052623 | Manual download from WDSI and Windows Update Catalog | Monthly |
| | Automatic update package | **definitionupdates.microsoft.com** | Monthly |
| **EDR sensor + features (server)** | Standalone package – KB5005292 (2012 R2 + 2016 only) | Windows Update Catalog | Monthly+ |

| Component | Method | Sources | Cadence |
|---|---|---|---|
| | Automatic update package | Windows Update | Monthly+ |
| **EDR sensor (client)** | Windows upgrade | Windows Update | Once yearly |
| **EDR features (client)** | Standalone package | Windows Update Catalog | Monthly+ |
| | Automatic update package | Windows Update | Monthly+ |

Table 7.1 – Update cadence

Note that whenever Windows Update is mentioned in the preceding table, those updates can be synced using a patch management solution such as **Windows Server Update Services** (**WSUS**)/ConfigMgr. They can be found under the **Definition Updates** category | **Software Update Point Component Properties** | the **Products** tab | **Windows Defender Antivirus** | **Microsoft Defender for Endpoint**. See *Figure 7.2* and *Figure 7.3* if you're unfamiliar with where to make that change in ConfigMgr.

From the **Administration** section, expand **Site Configuration**, and then choose **Sites**. Choose your site, then from the ribbon choose **Configure Site Components**, and select **Software Update Point**:

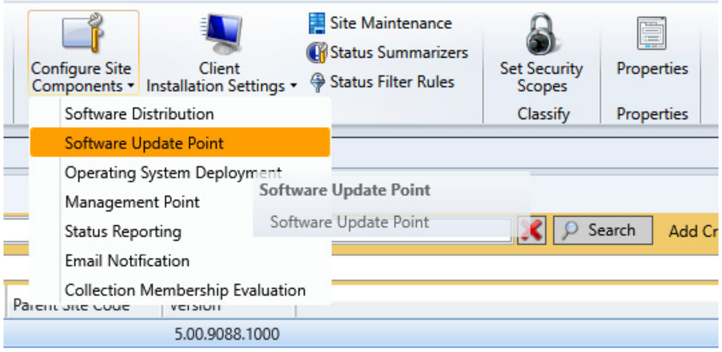
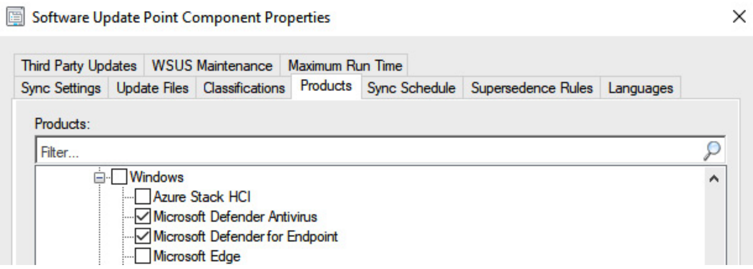


Figure 7.2 – SCCM site configuration

Scroll down to **Windows**, and under that, you'll see the two options mentioned previously to select and get synchronized:



- □ Windows
  - □ Azure Stack HCI
  - ☑ Microsoft Defender Antivirus
  - ☑ Microsoft Defender for Endpoint
  - □ Microsoft Edge

Figure 7.3 – Product selection for SUP

### Linux and macOS

**Security intelligence updates** for macOS and Linux are applied automatically and come from **definitionupdates.microsoft.com**.

Platform/app updates on macOS come from **Microsoft AutoUpdate** and will apply automatically if you haven't disabled it. On Linux, product updates come from the packages.microsoft.com repository and can be applied using your distribution's package manager – if you don't have a predefined update cadence for your Linux machines, you can consider scheduling updates using a cron job (the built-in Linux utility for scheduling processes to run at certain times).

For both OSs, the package comes with updates to both the antimalware and EDR components.

*COLD SNACK*

*If your Linux machines have been onboarded through Microsoft Defender for Cloud, the extension is reprovisioned regularly, which, if a new package is available, will trigger an upgrade. This reprovisioning occurs once a week.*

### Gradual rollout

MDE provides various update channels that essentially serve updates based on a device's participation. Leveraging configuration, you can exert some level of control over how early or late in Microsoft's rollout cycle you want devices to participate. This attempts to strike a balance between patch management (you control everything) and continuous updates (updates flow freely) by allowing you to build your own rings in your environment.

The following table shows the available channels for Windows and their equivalent (app/platform and engine updates always come as part of security intelligence on non-Windows) for macOS and Linux for *monthly* updates:

| Channel name | macOS and Linux | Description | Application |
|---|---|---|---|
| **Beta Channel - Prerelease** | **Beta/InsiderFast**<br>**insiders-fast** | Updates arrive here first – opt-in only. | Leverage this channel on specific machines where you want to consume changes first for testing purposes. On macOS and Linux, you will also get early access to new features this way. |

| Channel name | macOS and Linux | Description | Application |
|---|---|---|---|
| **Current Channel (Preview)** | **Preview/External insiders-slow** | Machines in this channel will be the first to receive updates when the release cycle starts. | Intended for pre-production/validation environments, as part of a safe deployment strategy – you will want at least a few machines in this channel. |
| **Current Channel (Staged)** | | Windows machines in this channel will receive updates later in the release cycle – essentially, once the preceding channel reaches 10% of the global population. | This is where you would explicitly place around 10% of your organization's machines – to make sure that you can identify a potential impact early on. |
| **Current Channel (Broad)** | **Current/Production production** | Machines assigned to this channel will receive updates at the end of the gradual cycle (typically, 3 weeks after it starts). | This will be the general population in your environment, if you wish to make the cycle more predictable. Otherwise, the default setting will automatically place your machine anywhere in the gradual cycle. |
| **Critical: Time Delay** | | Introduce a time delay. | This setting is intended for highly critical machines. You should still make sure to follow safe deployment principles and have some representative machines receive updates earlier. |

| Channel name | macOS and Linux | Description | Application |
|---|---|---|---|
| (default) | | Not configured. | The default setting will automatically set your machines to participate in the gradual process (**Preview-Staged-Broad**) |

Table 7.2 – Monthly update channels

*COLD SNACK*

*Default here means that your device will receive the update anywhere in the cycle from preview to broad (so it could be at the start, which is labeled* **Preview***, or the end, which makes it* **Broad***), which is typically a 3-week window.*

For *daily* security intelligence updates (released multiple times a day; there is no **Preview**, as the cadence is much too rapid), these are the available channels for Windows (macOS and Linux do not have separate options):

| Channel name | Description | Application |
|---|---|---|
| **Current Channel (Staged)** | **Get Current Channel updates later during gradual release** | This is where you would explicitly place around 10% of your organization's machines to make sure that you can identify the potential impact early on. |
| **Current Channel (Broad)** | **Get updates at the end of gradual release** | This setting is intended for highly critical machines. You should still make sure to follow safe deployment principles and have some representative machines receive updates earlier.<br>Note this setting will apply to both monthly and daily updates! |
| (default) | Not configured | The default setting will automatically set your machines to participate in the gradual process (**Staged-Broad**). |

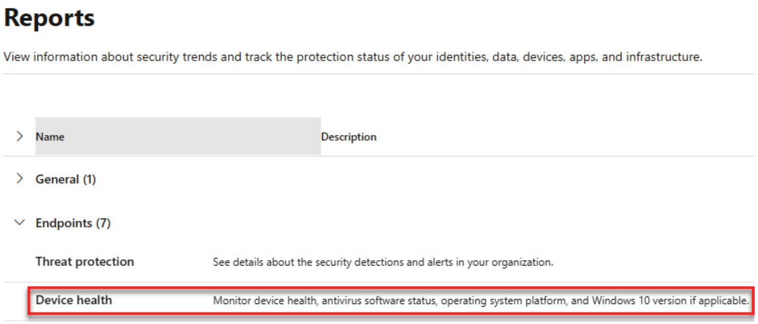
Table 7.3 – Daily update channels

You will want to stick to defaults for most of your devices. However, it makes a lot of sense to take early updates in part of your environment and delay in some other parts as part of a continuous safe deployment practice – the practice of rolling out gradually and to specific systems first, and monitoring the situation.

*Keeping up to date is a critical part of maintaining your security posture*. This not only applies to MDE itself but also to the OSs you are running and your applications. In addition, misconfiguration is a primary reason organizations are compromised. Let's look at how MDE can help you become more resilient through continuous security posture management.

## Maintaining security posture through continuous discovery and health monitoring

Now that we have our production deployments out in the environment, it's time to get on top of ensuring things are going well and that devices are healthy and functioning as expected. *Figure 7.5* shows the first report that we'll pull from in this section.



**Reports**

View information about security trends and track the protection status of your identities, data, devices, apps, and infrastructure.

| Name | Description |
|---|---|
| General (1) | |
| Endpoints (7) | |
| Threat protection | See details about the security detections and alerts in your organization. |
| Device health | Monitor device health, antivirus software status, operating system platform, and Windows 10 version if applicable. |

Figure 7.4 – MDE Reports dashboard

In the next few sections, we are going to select from some of the various areas in MDE where we will see the status of our EDR sensors, Defender Antivirus, and the settings we've deployed.

### Sensor health and operating system

The **Device Health** status report gives us a few pieces of information at a glimpse, such as the EDR sensor health, a chart of active, inactive, and impaired communications, and in case there's sensor data. We also get a breakdown of OSs and platforms, which gives us an idea of how healthy our environment is based on legacy versus modern OSs. *Figure 7.5* shows an example of the cards shown in this report.

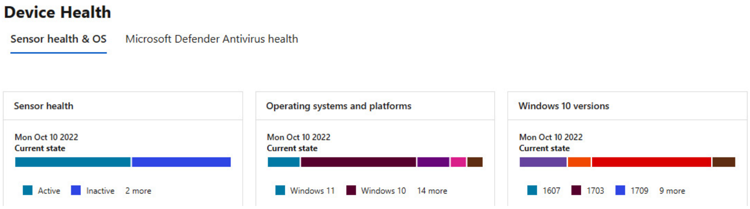
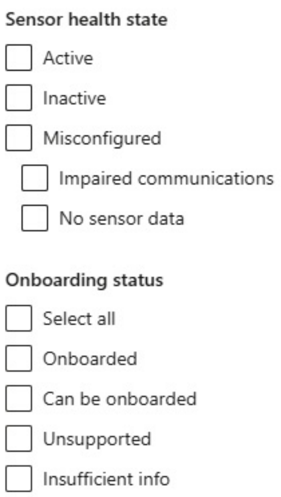


Figure 7.5 – The Device Health report page

Since the sensor health and OS health pages are just a high-level view, let's move on to the next section so that we can dive into these things a little deeper and cover some areas where you can view the health of the sensor.

### EDR sensor health state

There are a few ways we can look at EDR **sensor health** from the MDE portal. There is the high-level view from the sensor health and OS card described previously, as well as the **Devices** sub-node under the **Assets** node in MDE, which gives us an inventory of all devices currently onboarded, and also devices that can be onboarded (which we'll cover later). With filtering, as shown in *Figure 7.6*, we can also filter the **Sensor health state** setting. This is a great way to surface the devices having issues during onboarding. Perhaps they got the payload but they're behind a different firewall, or potentially they are in a DMZ. This is a perfect time to identify whether devices with those statuses are logically similar in the environment or one-offs. You can run the **Client Analyzer** tool to find out.



Sensor health state

☐ Active

☐ Inactive

☐ Misconfigured

☐ Impaired communications

☐ No sensor data

Onboarding status

☐ Select all

☐ Onboarded

☐ Can be onboarded

☐ Unsupported

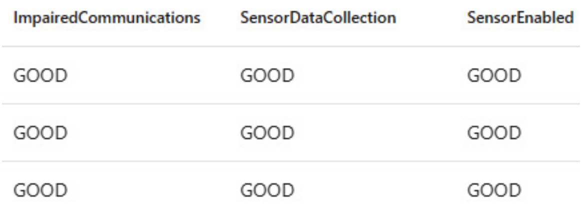
☐ Insufficient info

Figure 7.6 – Device asset filtering

Another way to dive into sensor health from the portal is through advanced hunting. Microsoft provides a query that you can find in the community queries called **Endpoint Agent Health Status Report**, under **General queries**, which looks like the following:

```
// This query will provide a report of many of the best practice configurations for Defender ATP
// This query was updated from https://github.com/Azure/Azure-Sentinel/tree/master/Hunting%20Quer
DeviceTvmSecureConfigurationAssessment
| where ConfigurationId in ('scid-91', 'scid-2000', 'scid-2001', 'scid-2002', 'scid-2003', 'scid
| extend Test = case(
    ConfigurationId == "scid-2000", "SensorEnabled",
    ConfigurationId == "scid-2001", "SensorDataCollection",
    ConfigurationId == "scid-2002", "ImpairedCommunications",
    ConfigurationId == "scid-2003", "TamperProtection",
    ConfigurationId == "scid-2010", "AntivirusEnabled",
    ConfigurationId == "scid-2011", "AntivirusSignatureVersion",
    ConfigurationId == "scid-2012", "RealtimeProtection",
    ConfigurationId == "scid-91", "BehaviorMonitoring",
    ConfigurationId == "scid-2013", "PUAProtection",
    ConfigurationId == "scid-2014", "AntivirusReporting",
    ConfigurationId == "scid-2016", "CloudProtection",
    "N/A"),
    Result = case(IsApplicable == 0, "N/A", IsCompliant == 1, "GOOD", "BAD")
| extend packed = pack(Test, Result)
| summarize Tests = make_bag(packed), DeviceName = any(DeviceName) by DeviceId
| evaluate bag_unpack(Tests)
```

If you run this, you see the health of some important features from both Defender Antivirus and the EDR sensor. *Figure 7.7* shows some of the sensor health-related items we can surface.



| ImpairedCommunications | SensorDataCollection | SensorEnabled |
|---|---|---|
| GOOD | GOOD | GOOD |
| GOOD | GOOD | GOOD |
| GOOD | GOOD | GOOD |

Figure 7.7 – A sample result from a community query

Play around with this query to return things of interest to you. Start looking for **BAD** results and go from there to see why those devices might be having issues.

So, from looking at some of the sensor-related items, we have the following:

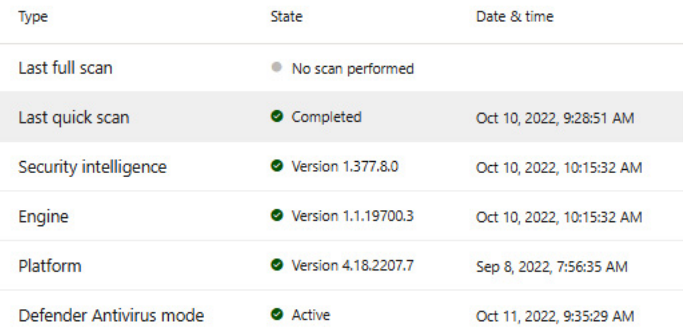
- **`ImpairedCommunications`**: This can mean a few different things; it is having issues talking out to the internet due to the necessary URLs not being open, or it may need a proxy. Keep in mind that the sensor requires **WinHTTP** (or Windows HTTP) to report sensor data to the backend cloud service.
- **`SensorDataCollection`**: The EDR sensor (**Sense**) is currently collecting and sending data back.
- **`SensorEnabled`**: **Sense** is onboarded and sending signals.

You can always peruse the **`Microsoft-Windows-SENSE/Operational`** logs in Windows Event Viewer on a case-by-case basis to see what communications are happening, or maybe not happening if you're troubleshooting.

Beyond checking the health of the sensor itself, it's also important to check the health and status of the settings that you have deployed. Now, for the EDR sensor, there are not that many settings to check, but nonetheless, it is worthwhile to discuss. You can check **`AMRunning mode`** with **`Get-MPComputerStatus`** in PowerShell; it would say **`EDR Block Mode`** if that was enabled. If it does, then **Defender Antivirus mode** shown in *Figure 7.8* (from the device page in the MDE portal) would say **EDR Block Mode** as well.

### Microsoft Defender Antivirus health

As with the EDR section, there are many ways to check various status-related entries for Defender Antivirus, one of which is the device entity page. The default dashboard shows the device health status for Defender Antivirus. *Figure 7.8* gives an example of what you would see there:



| Type | State | Date & time |
|---|---|---|
| Last full scan | ● No scan performed | |
| Last quick scan | ✓ Completed | Oct 10, 2022, 9:28:51 AM |
| Security intelligence | ✓ Version 1.377.8.0 | Oct 10, 2022, 10:15:32 AM |
| Engine | ✓ Version 1.1.19700.3 | Oct 10, 2022, 10:15:32 AM |
| Platform | ✓ Version 4.18.2207.7 | Sep 8, 2022, 7:56:35 AM |
| Defender Antivirus mode | ✓ Active | Oct 11, 2022, 9:35:29 AM |

Figure 7.8 – The device health status on the device entity page

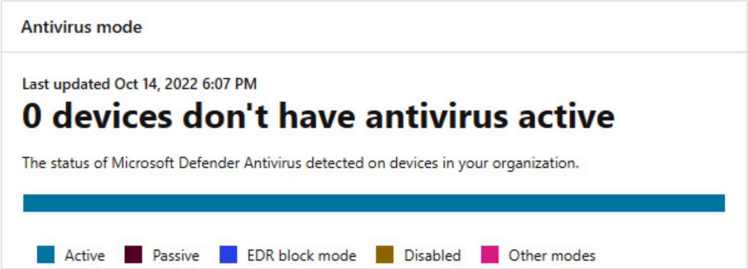
Looking at each of these, we can get the following quick view:

- **Last full scan**: This is the last time a last full scan was run and its status – whether it was manually invoked or scheduled
- **Last quick scan**: This is the last time a last quick scan was run and its status – whether it was manually invoked or scheduled
- **Security intelligence**: This gives you the current version of the security intelligence update and the date it was installed
- **Engine**: This gives you the current version of the antivirus engine and the date it was installed
- **Platform**: This gives you the current version of the antivirus platform and the date it was installed

Beyond the device entity page where we see the state and health of the individual machine, we have the Microsoft Defender Antivirus health report, found under the **Reports** node in MDE as a tab in the **Device health** report. This page gives us eight different cards that are all interactive; you can click the various bar graphs for fly-outs with additional information. Currently, the available cards are as follows:

- Antivirus mode card (**Active**, **Passive**, **EDR Block Mode**, **Disabled** or other modes – other could mean it's uninstalled or in an unexpected/error state)
- Antivirus engine version card
- Antivirus security intelligence version card
- Antivirus platform version card
- Recent antivirus scan results card
- Antivirus engine updates card
- Security intelligence updates card
- Antivirus platform updates card

*Figure 7.9* shows an example of a card from this dashboard. This card can be particularly helpful if you're migrating from a third-party antivirus, as you'll hopefully see your devices shifting from any of the non-active modes to active:



Antivirus mode

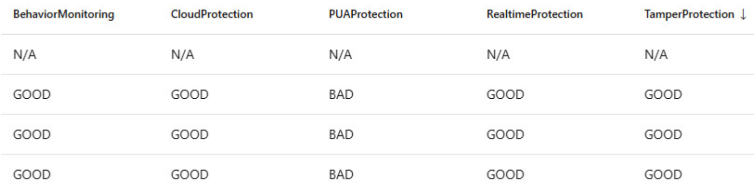
Last updated Oct 14, 2022 6:07 PM

**0 devices don't have antivirus active**

The status of Microsoft Defender Antivirus detected on devices in your organization.

Figure 7.9 – A card from the Microsoft Defender Antivirus health report

Referencing the same preceding query, from advanced hunting, you can see a host of checks for Defender Antivirus beyond some of the information you've already revised at this point. See *Figure 7.10* for a sample of this:



| BehaviorMonitoring | CloudProtection | PUAProtection | RealtimeProtection | TamperProtection ↓ |
|---|---|---|---|---|
| N/A | N/A | N/A | N/A | N/A |
| GOOD | GOOD | BAD | GOOD | GOOD |
| GOOD | GOOD | BAD | GOOD | GOOD |
| GOOD | GOOD | BAD | GOOD | GOOD |

Figure 7.10 – A sample result from a community query

Here are some of the other items you can see:

- **TamperProtection**
- **RealtimeProtection**
- **BehaviorMonitoring**
- **PUAProtection**
- **CloudProtection**

That wraps up some of the reporting available that would be most valuable as you start your production deployments.

### ASR monitoring

When it comes to monitoring for ASR, it's really just ASR rules that get their own report (at least so far). CFA and NP are really something that you would look at in advanced hunting if you were interested in seeing those events. Refer to the *Performing production readiness checks* section earlier in this chapter for some queries to get you started.

Monitoring for ASR rules has improved greatly recently; you get a wonderful view of the detections, the breakdown of your configurations across all devices, as well as a list of filenames detected by rules. Let's break each of these down briefly and show a little of what they're about.

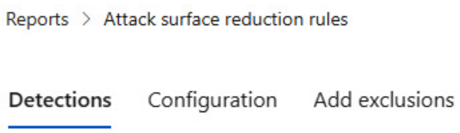
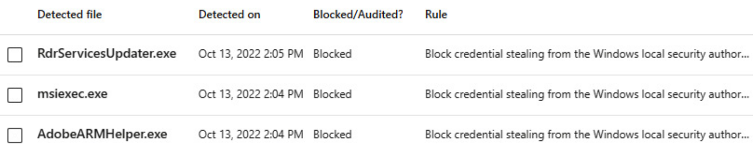


Figure 7.11 – The ASR rules report tabs

At the top of the **Detections** report, you get some traditional bar graphs that show the amount of detection, whether that's audit or block events.

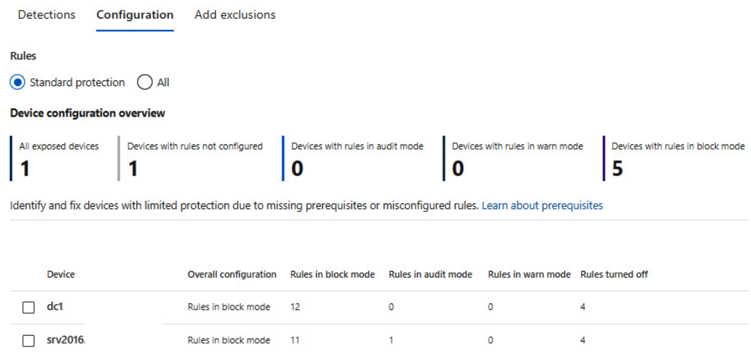
The following pane shows the most recent events in date form, as shown in *Figure 7.12*. When looking at the detections page for all ASR rules, you are presented with many details, such as the source app, device, device group, user, and publisher. You can even group the detection bases on these attributes too, making them easier to summarize.



| | Detected file | Detected on | Blocked/Audited? | Rule |
|---|---|---|---|---|
| □ | RdrServicesUpdater.exe | Oct 13, 2022 2:05 PM | Blocked | Block credential stealing from the Windows local security author... |
| □ | msiexec.exe | Oct 13, 2022 2:04 PM | Blocked | Block credential stealing from the Windows local security author... |
| □ | AdobeARMHelper.exe | Oct 13, 2022 2:04 PM | Blocked | Block credential stealing from the Windows local security author... |

Figure 7.12 – The ASR rule detection report

Next up in the ASR rule report is **Configuration**, which gives you a great high-level view of what the device status is when it comes to all available rules. The overview gives you the holistic numbers of devices and their configurations, and the lower pane shows you each individually. When you select a device, a fly-out shows you the status of each rule on that device:



Rules

● Standard protection ○ All

Device configuration overview

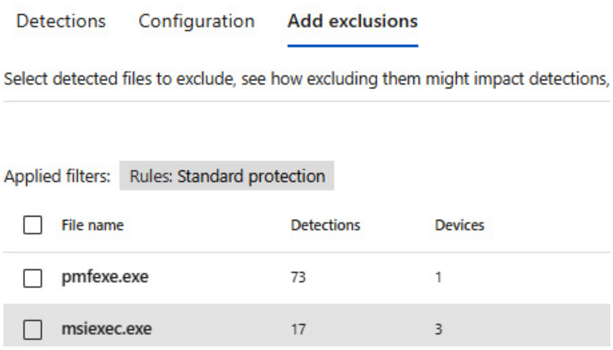
| All exposed devices | Devices with rules not configured | Devices with rules in audit mode | Devices with rules in warn mode | Devices with rules in block mode |
|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 5 |

Identify and fix devices with limited protection due to missing prerequisites or misconfigured rules. Learn about prerequisites

| | Device | Overall configuration | Rules in block mode | Rules in audit mode | Rules in warn mode | Rules turned off |
|---|---|---|---|---|---|---|
| □ | dc1 | Rules in block mode | 12 | 0 | 0 | 4 |
| □ | srv2016. | Rules in block mode | 11 | 1 | 0 | 4 |

Figure 7.13 – The ASR rule configuration report

Rounding out the reporting on ASR rules, we have the **Add exclusions** tab. This provides some great information on the detections per device by file, and when you select one, you get the fly-out pane again. There, you can look at the total amount of detections versus how many you can see after the exclusion of said file. We suggest you read the ASR FAQ at **https://learn.microsoft.com/en-us/microsoft-365/security/defender-endpoint/attack-surface-reduction-faq** before you start adding exclusions so that you can understand how some rules make decisions:



Select detected files to exclude, see how excluding them might impact detections,

Applied filters: Rules: Standard protection

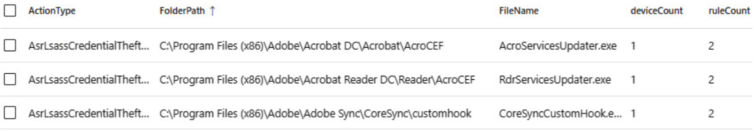
| □ | File name | Detections | Devices |
|---|---|---|---|
| □ | pmfexe.exe | 73 | 1 |
| □ | msiexec.exe | 17 | 3 |

Figure 7.14 – The ASR rule exclusion report

The last way to check on some ASR rule detections once you have them deployed is by using advanced hunting. The following is a sample you can get started with:

```
//(1) Get FolderPath, FileName, deviceCount and ruleCount for "Block credential stealing from th
DeviceEvents
| where ActionType contains 'AsrLsassCredentialTheft'
| distinct ActionType, FolderPath, FileName, DeviceId
| summarize deviceCount = count() by ActionType, FolderPath, FileName
| join (DeviceEvents| summarize ruleCount = count() by ActionType, FolderPath, FileName) on $lef
| project ActionType, FolderPath, FileName, deviceCount, ruleCount
| order by ActionType, ruleCount desc
```

Refer to the following screenshot, which shows some sample output:



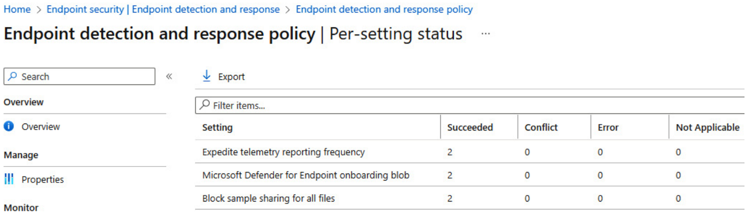
| ☐ | ActionType | FolderPath ↑ | FileName | deviceCount | ruleCount |
|---|---|---|---|---|---|
| ☐ | AsrLsassCredentialTheft... | C:\Program Files (x86)\Adobe\Acrobat DC\Acrobat\AcroCEF | AcroServicesUpdater.exe | 1 | 2 |
| ☐ | AsrLsassCredentialTheft... | C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroCEF | RdrServicesUpdater.exe | 1 | 2 |
| ☐ | AsrLsassCredentialTheft... | C:\Program Files (x86)\Adobe\Adobe Sync\CoreSync\customhook | CoreSyncCustomHook.e... | 1 | 2 |

Figure 7.15 – Output from the ASR detection query

The MDE portal offers some effective options. However, both Intune and ConfigMgr have their own reporting for prevention capabilities; they may provide more operational-focused insights.

## Intune reports

Reporting in Intune on the EDR side is going to be entirely about the success of the deployment and the three main settings you define with it. Those settings include the success of the onboarding blob itself, whether you chose to expedite telemetry or not, and sample sharing. The following screenshot gives you a quick view of the status of these settings:

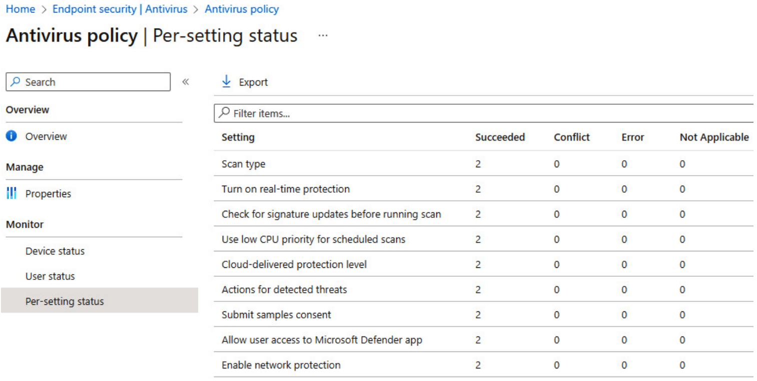


Home > Endpoint security | Endpoint detection and response > Endpoint detection and response policy
Endpoint detection and response policy | Per-setting status ···
Search
«
Overview
Overview
Manage
Properties
Monitor
Export
Filter items...

| Setting | Succeeded | Conflict | Error | Not Applicable |
|---|---|---|---|---|
| Expedite telemetry reporting frequency | 2 | 0 | 0 | 0 |
| Microsoft Defender for Endpoint onboarding blob | 2 | 0 | 0 | 0 |
| Block sample sharing for all files | 2 | 0 | 0 | 0 |

Figure 7.16 – The Intune report on EDR deployment

And, of course, you can head over to the **Endpoint security** blade to view how some of your antivirus settings are doing. The next screenshot again gives you a glimpse of what you can expect from that policy report:



**Antivirus policy** | Per-setting status

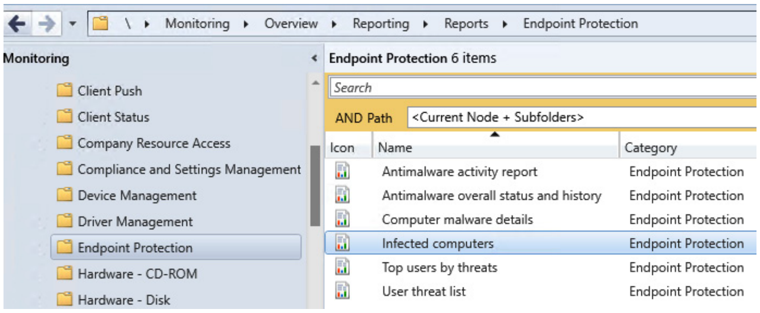
| Setting | Succeeded | Conflict | Error | Not Applicable |
|---|---|---|---|---|
| Scan type | 2 | 0 | 0 | 0 |
| Turn on real-time protection | 2 | 0 | 0 | 0 |
| Check for signature updates before running scan | 2 | 0 | 0 | 0 |
| Use low CPU priority for scheduled scans | 2 | 0 | 0 | 0 |
| Cloud-delivered protection level | 2 | 0 | 0 | 0 |
| Actions for detected threats | 2 | 0 | 0 | 0 |
| Submit samples consent | 2 | 0 | 0 | 0 |
| Allow user access to Microsoft Defender app | 2 | 0 | 0 | 0 |
| Enable network protection | 2 | 0 | 0 | 0 |

Figure 7.17 – The Intune report on antivirus settings

Intune does not break out the per-setting status for each ASR rule, so there is nothing much to show there in case you're wondering. Alright, let's move on to some reports in ConfigMgr.

## ConfigMgr reports

There are a few nice reports in ConfigMgr that show similar details to what MDE and Intune do within their respective reports. The following screenshot shows the reports that are there out of the box, and if you're good at working with **Microsoft Report Builder**, you can take these and mold them into other reports, pulling in additional information or re-forming how they are as is:



**Endpoint Protection** 6 items

| Icon | Name | Category |
|---|---|---|
| | Antimalware activity report | Endpoint Protection |
| | Antimalware overall status and history | Endpoint Protection |
| | Computer malware details | Endpoint Protection |
| | Infected computers | Endpoint Protection |
| | Top users by threats | Endpoint Protection |
| | User threat list | Endpoint Protection |

Figure 7.18 – The ConfigMgr report for endpoint protection

That sums up the high-level reporting that can be found between the various first-party applications. In the next section, we'll take a look at the threat and vulnerability management telemetry that can now be surfaced to you.

# Getting started with vulnerability management

Now that we have MDE deployed in our environment, it's time to observe some of the results. One of the big areas where we're going to see new information is in vulnerability management, where we'll see everything from what our EDR is scanning and relaying back for us in terms of posture improvements to additional recommendations that are available for you to roll out, or even just generally where the weaknesses are that need improvement.

Assessing vulnerabilities within an environment can be a daunting, even overwhelming, task to overcome, in most cases requiring multiple skill sets, tools, and potentially a significant headcount to stay ahead of. The vulnerability management node in MDE succeeds not only at providing a clear overview of risk in your environment but also at providing you with the insight to develop strategic plans to mitigate these risks.

Built upon expert-level threat monitoring and analysis, vulnerability management is an add-on or standalone product that augments your MDE experience by assessing risk in your environment and giving you actionable feedback. This is accomplished through asset discovery and inventory capabilities, continuous vulnerability, misconfiguration assessment, security baseline assessment, prioritized security recommendations, and seamless remediation and progress tracking. You can even directly block vulnerable applications if needed.

We will not exhaustively cover vulnerability management here but want to provide a comprehensive high-level overview of the features as an introduction.

## Dashboard

The first sub-node within vulnerability management, the dashboard, gives you a robust high-level view of your environment. This has cards scoring risk levels such as **Exposure score** and **Microsoft Secure Score for Devices**, as well as cards that show highlights in your environment with **Top remediation activities**, **Top vulnerable software**, **Top exposed devices**, **Top security recommendations**, and **Top remediation activities**. All of these things combined can help to quickly gauge what needs attention or what will have the greatest impact.

## Security recommendations

When it comes to device maintenance, there are some things you really want to focus on:

- Are OSs approaching or at **end of life** (**EOL**)?
- What is the patch level of devices in the environment?
- Are there drivers or firmware that are vulnerable?
- Are the applications in your environment up to date or vulnerable?

Let's dive into this from the point of view of **Microsoft Defender Vulnerability Management** (**MDVM**). Some of the things we'll show in this section might be tasks that go to different teams, depending on how large your organization is and how responsibilities are distributed. We're trying to be agnostic of roles, as we know many of you will have varying team sizes, some wearing many hats.

The second sub-node under vulnerability management is **Security recommendations**, with the node name in the menu shortened to simply **Recommendations**. This node provides an overview of all security recommendations that MDE has for your environment, quantified and qualified so that you can quickly understand what to focus on. At the top are cards to highlight, especially important recommendations, such as discovered devices that are not onboarded to MDE or devices that are exposed to very recent, high-risk vulnerabilities. Most of the columns are self-explanatory, but there are a few things to highlight:

- Security recommendations range from system configuration to firmware and software updates, OS upgrades, or even software uninstalls.
- Weaknesses are a direct correlation to the **Weaknesses** sub-node of vulnerability management and are representative of your exposure to the security gap the recommendation is related to.
- Indications of active alerts or known public exploits for **Common Vulnerabilities and Exposures** (**CVEs**) or recommendations can be found via the **Threats** column. This column and icon are common throughout vulnerability management, so it's good to understand what they mean on the fly. **Threats** has two icons – a target and a bug. The target represents active alerts in your environment, and the bug represents known public exploits. If either is present in your environment for that vulnerability, the appropriate icon will be highlighted in red. These details are shown in *Figure 7.19*.

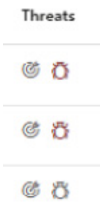
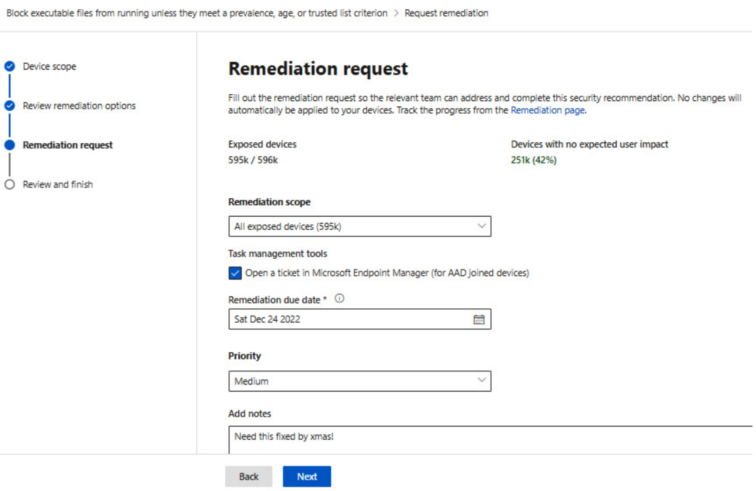


Figure 7.19 – MDVM threat symbols

- The last column that warrants description is the **Impact** column. **Impact** quantifies the change that fulfilling the recommendation will have on your exposure and secure scores. If **Impact** shows a triangle pointing downward, it indicates an exposure reduction. If it has a plus symbol (+), it indicates an increase to secure the score. Some recommendations will improve both scores.

If you select any given recommendation, you will get a fly-out with more details on that recommendation (we'll cover a few examples of application and OS recommendations further down). The reasoning behind the recommendation is given on the **General** tab. In the case of software updates, this may be as simple as bulleted explanations of relevant exploits that warrant the change alongside CVE and exposure counts. With block recommendations, you not only get a detailed explanation of why the block is recommended but also telemetry-based insights into what systems you can enable the block on, with little to no expected risk to user productivity.

At the bottom of the fly-out, you will see options to request remediation and create exceptions as needed. In both cases, you can target a particular MDE device group or all relevant MDE devices as needed. You can also pivot to the relevant software page via the **Open software page** button at the top. If there are multiple related software (common if it's an OS-based finding), you will be able to pick the appropriate one you are interested in from a drop-down menu:



Block executable files from running unless they meet a prevalence, age, or trusted list criterion > Request remediation

- Device scope
- Review remediation options
- **Remediation request**
- Review and finish

**Remediation request**

Fill out the remediation request so the relevant team can address and complete this security recommendation. No changes will automatically be applied to your devices. Track the progress from the Remediation page.

| Exposed devices | Devices with no expected user impact |
|---|---|
| 595k / 596k | 251k (42%) |

**Remediation scope**

All exposed devices (595k)

Task management tools

☑ Open a ticket in Microsoft Endpoint Manager (for AAD joined devices)

Remediation due date *

Sat Dec 24 2022

**Priority**

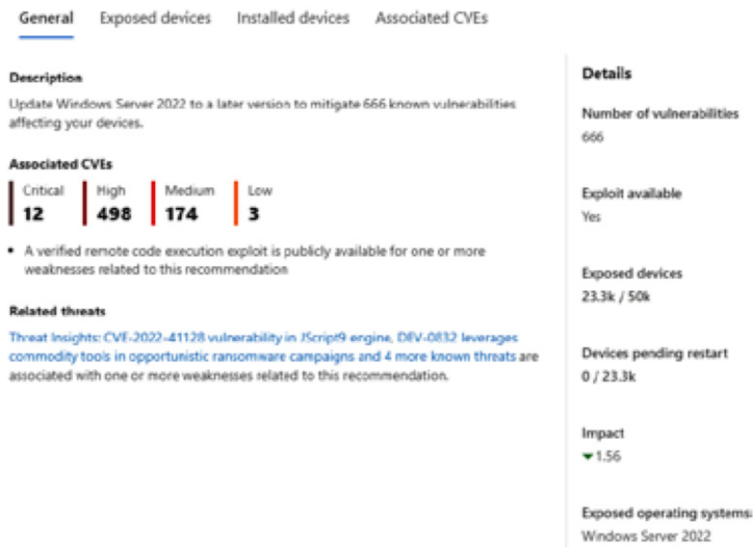
Medium

Add notes

Need this fixed by xmas!

Figure 7.20 – A remediation request example

From an MDE standpoint, specifically security recommendations under vulnerability management, we get a different view on the importance of keeping on top of system health versus what we talked about in ***Chapter 5***, *Planning and Preparing for Deployment*. What's meant by that is it is no longer just about needing to plan migrations because there are newer OSs out there; instead, it's about Server 2019 being vulnerable to *x* vulnerabilities, so let's think about getting those upgraded. *Figure 7.21* gives an example of this:



**Description**

Update Windows Server 2022 to a later version to mitigate 666 known vulnerabilities affecting your devices.

**Associated CVEs**

| Critical | High | Medium | Low |
|---|---|---|---|
| 12 | 498 | 174 | 3 |

- A verified remote code execution exploit is publicly available for one or more weaknesses related to this recommendation

**Related threats**

Threat Insights: CVE-2022-41128 vulnerability in JScript9 engine, DEV-0832 leverages commodity tools in opportunistic ransomware campaigns and 4 more known threats are associated with one or more weaknesses related to this recommendation.

**Details**

Number of vulnerabilities
666

Exploit available
Yes

Exposed devices
23.3k / 50k

Devices pending restart
0 / 23.3k

Impact
▼1.56

Exposed operating systems
Windows Server 2022

Figure 7.21 – Security recommendations for OS upgrades

*COLD SNACK*

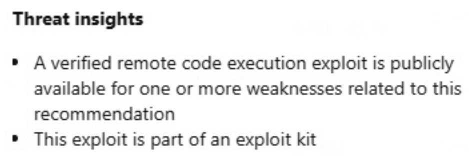
*For various vulnerabilities, Microsoft will share any related threats that it has on a particular usage it has observed. In Figure 7.21, Microsoft is sharing information on an exploit with the* **Microsoft Support Diagnostic Tool** *(***MSDT***), where an application such as Microsoft Word runs and calls MSDT, which can then run arbitrary code to do things such as installing another app, changing or even deleting data, or even creating new accounts.*

Keeping the OS healthy means ensuring it is getting the latest upgrades or updates and that the applications residing on it also keep up with the latest changes. Often, older legacy OSs are kept afloat because an application in use does not work on anything modern. We have all heard the justification: "*Yeah, we still have Server 2008 because of application x, and the person that wrote it left 10 years ago.*" This is, of course, easier said than done, but regardless, a business decision should be made on moving past that excuse.

After understanding the importance of keeping the OS healthy and up to date, we need to consider the upkeep of the applications that reside on it. When looking at the attack surface, it can be a lot of work just to get the OS in order, not to mention all the applications. Again, this is easier said than done. Quite frankly, this is one of the biggest things we see companies struggling with.

Application management and shadow IT are challenging, but MDE is here to help with threat and vulnerability management. While it's not set up as an intentional asset inventory or asset management system, it does help with the scanning aspect in that it will inventory each device to the best of its ability.

*Figure 7.22* gives you an example of why it's imperative to get applications under management because, at any given point, there can be a publicly available exploit. Take these as opportunities to say, hey, do we manage this app properly? How do we even manage it? Should this app even be allowed in an environment? If you happen to also use **Microsoft Defender for Cloud Apps** (**MDCA**), you can leverage the ability to sanction and un-sanction apps and actually have MDE block them on endpoints.



**Threat insights**

- A verified remote code execution exploit is publicly available for one or more weaknesses related to this recommendation
- This exploit is part of an exploit kit

Figure 7.22 – Security recommendations for application updates

As far as additional tabs go, software and firmware updates will give you a list of exposed and compliant devices, as well as links to the relevant CVEs. Blocks will outline distinct remediation options on a separate tab and list exposed devices on another.

As mentioned in *Chapter 4*, *Understanding Endpoint Detection and Response*, device entities have a tab for **Security recommendations**. The content there is the same, just specific to the given device.

## Remediation

The **Remediation** sub-node of vulnerability management shows all active remediation activities, exceptions, and blocked applications. This is a direct reflection of remediation and exception requests made through the **Security recommendations** node.

## Inventories

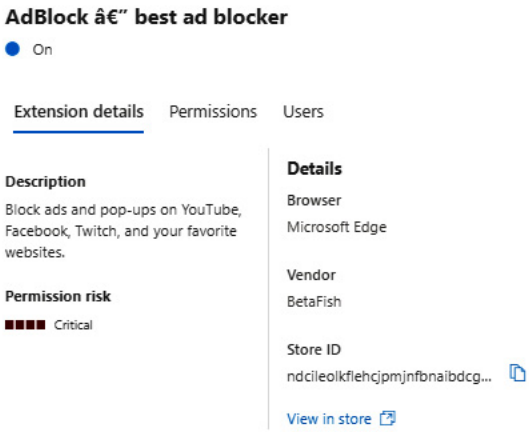
Encapsulated under this sub-node of vulnerability management are all the inventories MDE has for onboarded devices. As with security recommendations, device pages have a tab for **Software inventory**, **Browser extensions**, and **Certificate inventory**. The content there is the same as what's described here, just specific to the device rather than aggregated for the environment.

### Software inventory

This tab gives you a list of software installed on devices in your environment and relevant details about that software. As always, selecting an individual item will get you a fly-out with more information and allow you to drill down into what devices the software is installed on.

### Browser extensions

This provides statistics and descriptions for all browser extensions in your environment. This is all self-explanatory, but what is especially helpful to call out here is the qualification of risk associated with the permissions the extension requires, and that the details have the store ID as well as a link to the extension in the relevant store. You will find a tab in the fly-out that lets you see what user installed an extension on any given device, which is especially useful when you're navigating to the browser extension information from a device entity page:



**AdBlock â€" best ad blocker**

● On

Extension details | Permissions | Users

**Description**

Block ads and pop-ups on YouTube, Facebook, Twitch, and your favorite websites.

**Permission risk**

■■■■ Critical

**Details**

Browser
Microsoft Edge

Vendor
BetaFish

Store ID
ndcileolkflehcjpmjnfbnaibdcg...

View in store

Figure 7.23 – The browser extension fly-out

### Certificates

The **Certificates** tab allows you to access the certificate inventory from a device's page. This tab is fantastic for reviewing certificates, as it gives you a good look into the details of each one, such as the name, expiration and issue dates, the key size, who issued it, the signature algorithm, key usage, and which devices it resides on. Expired certificates will show a little red triangle with an exclamation mark (bang) icon for easy identification.

### Firmware

This inventories tab gives visibility into known BIOS and processor exploits based on advisories provided by **Lenovo**, **Dell**, and **Hewlett-Packard** (**HP**) exclusively (at the time of writing). It will explicitly tell you in the portal that the status of even the same vulnerabilities on other vendor devices is unknown.

From here, you can use the high-level metrics to prioritize and then pivot through the relevant firmware update security recommendations.

## Weaknesses

This is where all CVEs within MDE are stored. Similar metrics to all the other aspects of vulnerability management exist here and also help to quantify risk within your estate. The fly-out contains details on the vulnerability, whether it has a known exploit, and the status of the relevant security updates within your environment. You can also pivot through deep links to exposed devices and related software.

### Discovered vulnerabilities

The device page equivalent of **Weaknesses**, this tab displays a list of vulnerabilities a device is susceptible to, including the relevant CVE number, severity, **Common Vulnerability Scoring System** (**CVSS**) score, what software is impacted (by the CVE entirely, not specific to the given device), when the CVE was published, the first detected and updated timestamps, and any relevant tags.

## Event timeline

Not to be confused with the timeline on device pages, the event timeline acts as a news feed for your organization to explain how risk is introduced through new exploits and vulnerabilities. This is the best place to check whether you have a drastic change in your exposure or secure scores. In this sub-node, you will get information on any fundamental change to a vulnerability. For example, you will get updates on new vulnerabilities, when an unexploited vulnerability becomes exploited, and when an exploit is added to an exploit kit.

### Security baselines assessment

A security baseline is a foundational set of configurations that you apply to a system or group of systems that ensures they meet a certain standard for security hardening. This can be driven by a need to meet regulatory compliance requirements, to ensure crown jewels are protected, or even just to create a baseline for your standard end user clients to improve their security posture.

*COLD SNACK*

*Defined by the* **Center for Internet Security** *(***CIS***), the available controls are pulled from many well-known standards such as NIST Cybersecurity Framework, ISO 27000, and also PCI DSS and HIPAA.*

The **Security baseline assessment** sub-node of vulnerability management, which is abbreviated as **Baselines assessment**, provides a continuous and effortless way to monitor compliance with security baselines in your organization. From here, you can quickly create profiles based on benchmarks derived from recommendations by Microsoft, CIS, or the **Defense Information Systems Agency's (DISA's) Security Technical Implementations Guides (STIGs)**. These profiles aren't locked though. Once you choose your scope, you can customize any of the populated configurations to fit the needs of your organization.

## Summary

This chapter has provided more insights into what it takes for a successful production rollout, how to maintain your installations, and how to improve your overall security posture. You've also been introduced to the features of vulnerability management.

Operational excellence is how you optimize your investments and stay ahead of the security game – it takes time, patience, and continuous attention, but it's worth it. With what you've learned in this chapter, you now know what areas to focus on and what options you have to be successful.

*COLD SNACK*

*If, like many in the infosec community, you are following* `@SwiftOnSecurity`*, you will notice that the theme of operational excellence makes a frequent appearance in their content. The authors of this book, coming from a variety of backgrounds, can only emphasize how important this is!*

In the next chapter, we'll walk through practical examples of how to analyze and respond to threats surfaced by these tools that you've worked so hard to implement.