

# Patch Management ist keine Raketen Wissenschaft mehr

Meine Erfahrungen und einfache Schritte

Jürgen Rinelli



Jürgen Rinelli  
Am Eichel 6a  
85302 Alberzell  
[info@success.eu.com](mailto:info@success.eu.com)

## Inhaltsverzeichnis

1. Ein paar Worte	1
2. Schritt für Schritt	2
2.1 Automatisiertes Test und Freigabeverfahren:	2
2.2 Automatische Verteilungen einrichten:	8
2.3 Überprüfung weiterer Schwachstellen im Thread Management:	15
2.4 Zeit für den Cappuccino – Dashboards prüfen:	18
3. About the Author:	22

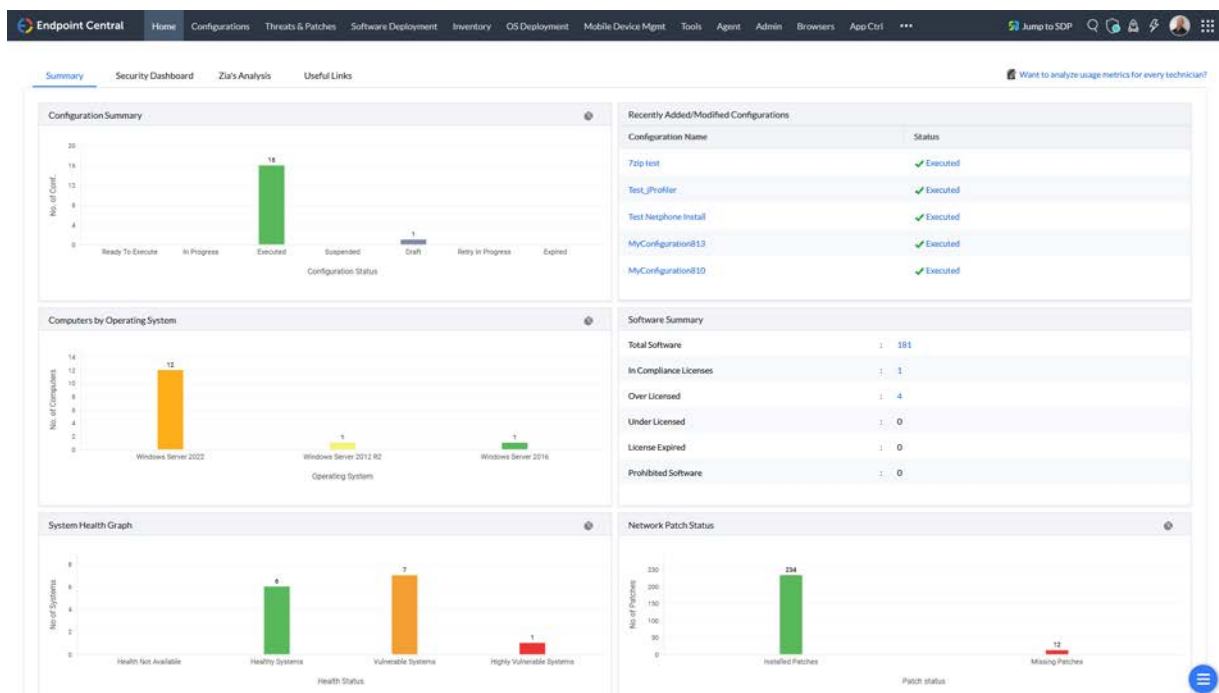
## 1. Ein paar Worte

Als Senior Consultant sehe ich immer wieder Firmen, die gehackt wurden, und unterstütze diese beim Neuaufbau. Meist ist der Grund eine unzureichende oder nicht vorhandene Endpoint-Management-Strategie. Was früher noch mit viel Aufwand einherging, ist heute keine Raketenwissenschaft mehr.

Bei der Vielzahl an Cyberthreads und durch Nutzung von KI Technologien der Angreifer, werden Attacken immer intelligenter, schneller und anpassungsfähiger.

In diesem kleinen E-Book teile ich meine Best Practices für das Endpoint-Patch-Management und zeige wie einfach eine umfangreiche Patch-Strategie eingerichtet werden kann. Meine persönlich bevorzugte Lösung und Grundlage auf welcher die Screenshots und Einstellungen basieren, ist von ManageEngine.

Ich nutze und empfehle die EndpointCentral Security Edition, wobei meine Best Practices bezogen auf das reine Patchmanagement dieses mini E-Books, auch mit dem VulnerabilityManager Plus angewendet werden können.



„Patch-Management“ und „Endpoint Security“ sind Begriffe, welche sich heute keinesfalls mehr trennen lassen. Ich erzähle auch nichts Neues, wenn ich sage, dass es die Endgeräte sind, welche am anfälligsten für Cyberattacken sind. **Ale** Endgeräte – worunter ich auch die User zähle!

Während ich bei Usern ein „Patch-Management“ nur über konsequente Schulungen durch gefakte E-Mails, Links, etc. durchführen kann – und sollte, lässt sich Hardware mit deutlich geringerem Widerstand und Aufwand patchen.

Was ist für mich wichtig und welche Schritte gehe ich beim Patch-Management?

1. Ich richte ein automatisches Test und Freigabeverfahren ein.
2. Ich richte **zwei** automatische Verteilungen der freigegebenen Patches ein.
3. Ich prüfe, welche Schwachstellen noch offen sind im Threat-Management und behebe diese.
4. Ich prüfe regelmäßig die Dashboards bei einer guten Tasse Cappuccino.

Nach diesem ersten einfachen Schritt der Patch-Automatisierung durch automatisches Testen und anschließender automatischer Verteilung, ist eine Grundsicherheit der Geräte gegeben. Das damit die Endpoints nicht komplett gesichert sind, habe ich schon erwähnt. Es gilt auch biologische Endpoints (User) an die Hand zu nehmen. Auf diesen Punkt gehe ich in einem separaten E-Book mit den Themen Device-Control, Applikation-Control, Data-Loss-Prevention, Anti-Ransomware und ganz wichtig – Browser Security ein.

Sollen wir loslegen?

## 2. Schritt für Schritt

### 2.1 Automatisiertes Test und Freigabeverfahren:

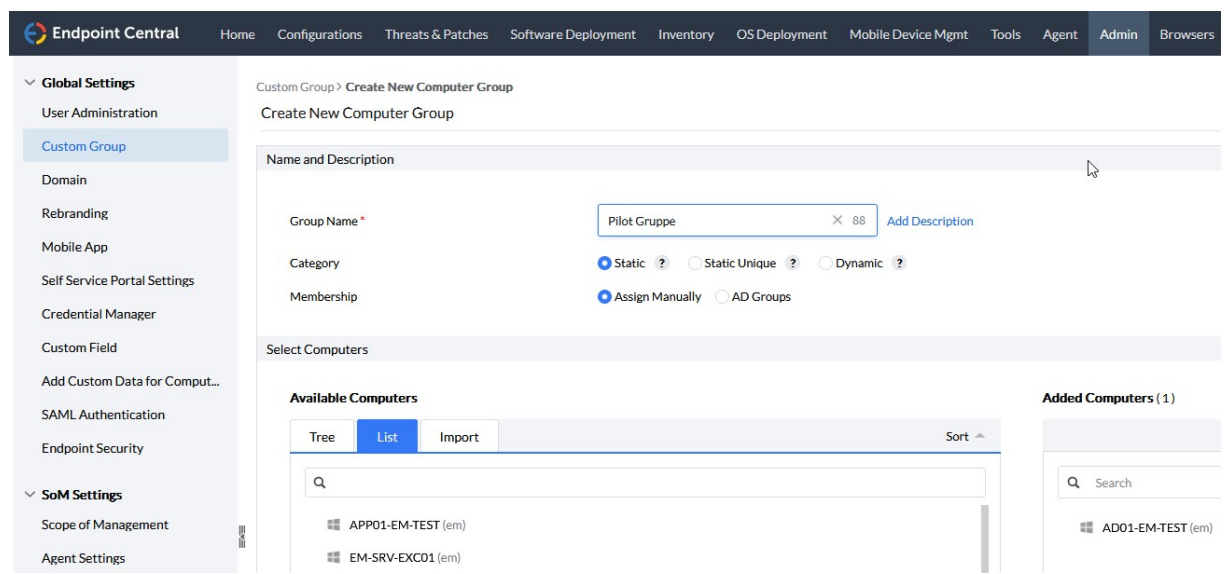
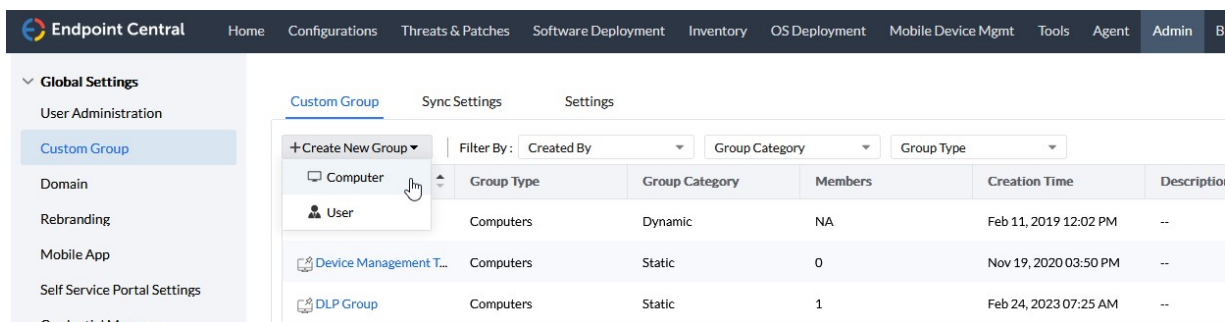
Da ein Patch auch zu unerwünschtem Verhalten führen kann, gebe ich Patches nur in den seltensten Fällen ungeprüft raus. Ja – besser wäre es, alles zu testen!

Um mir die Arbeit einfacher zu machen, gebe ich das Testen ab. Hierzu erstelle ich zuerst eine Testgruppe mit meinen eigenen Testgeräten sowie auch Geräten von Key-Usern. Also User aus jeder Abteilung, welche mit der Software täglich arbeiten. Mir ist es oft aus Lizenz und Berechtigungsgründen nicht möglich, Aktionen in der Software auszuführen. Oft kann ich die Software überhaupt nicht auf meinen Testgeräten installieren, wegen fehlender Lizenz. Also benötige ich User mit dieser Software welche auch Aktionen in dieser ausführen. Nur so kann sichergestellt werden das ein weiterführender Link z.B. zu einer Reader-Applikation, auch funktioniert.

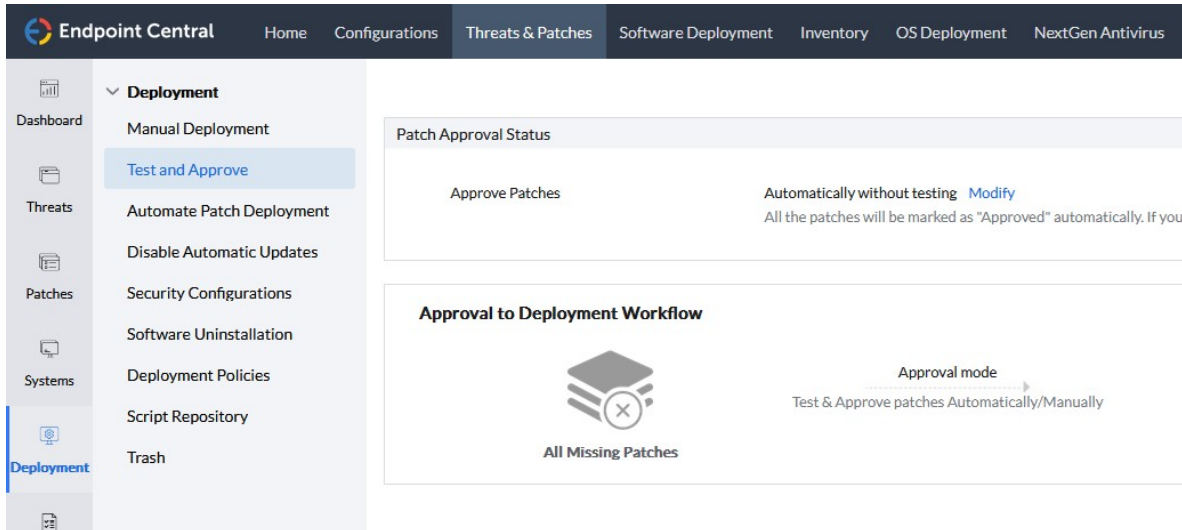
Bei den Key-Usern binde ich vor allem die User ein welche sich bisher auch schon häufig mit Fehlern oder langsamen Systemen gemeldet haben. – Ich will ja Feedback und den möglichst schnell!

Wie eingangs bereits erwähnt zeige ich die Einrichtungsschritte exemplarisch an meinem Favoriten dem EndpointCentral UEMS von ManageEngine, denn einfacher und intuitiver habe ich es bisher in noch keiner anderen Endpoint Management Lösung auf dem Markt gefunden – und ich habe mit vielen gearbeitet in den letzten 25 Jahren.

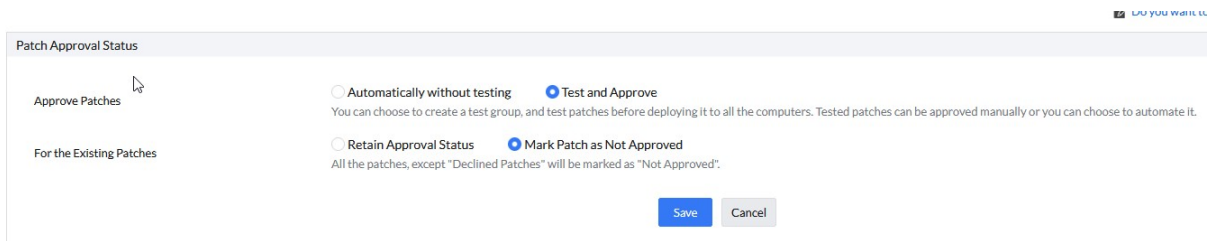
Zuerst erstelle ich eine statische Computergruppe über Admin/Custom Group und packe alle Testsysteme und Key-User Systeme hinein.



Weiter gehts in Threads & Patches/Deployment/Test and Approve, um ein automatisches Test- und Freigabeverfahren einzurichten.



Dort stelle ich – falls noch nicht geschehen – den Patch Approval Status über den Modify-Button von „Automatically without testing“ um auf „Test and Approve“ und „Mark Patch as not Approved“ um die bereits automatisch freigegebenen Patches sicherheitshalber auf nicht freigegeben, zu setzen.



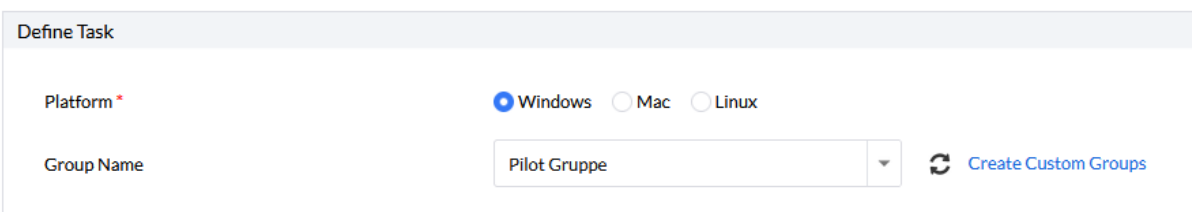
Nach dem Speichern der Einstellung kann ich über „AddGroup“ meine erste Testgruppe anlegen. Es können verschiedene erstellt werden für unterschiedliche Szenarien. Ich trenne Clients und Server auf jeden Fall voneinander. Ja – auch Server können zu einem gewissen Teil automatisch gepatcht werden.

Im ersten Schritt wähle ich die Plattform aus: Windows, Mac oder Linux

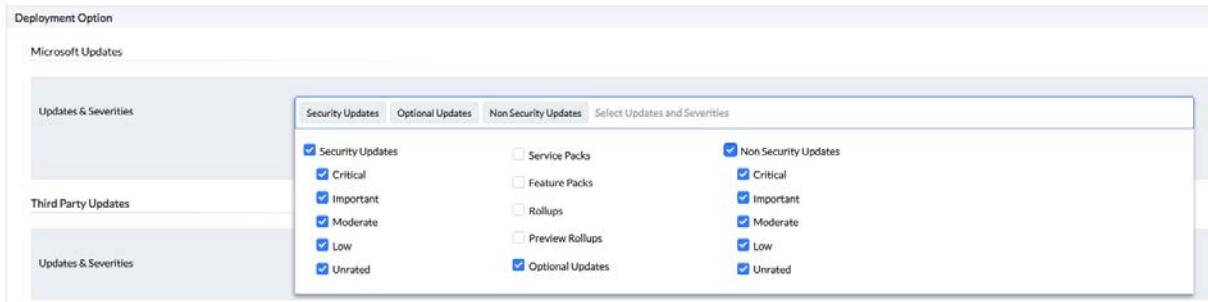
Dann wird die zuvor erstellte Gruppe benötigt – „Pilot Gruppe“ in diesem Beispiel.

Test and Approve > Test group settings

### Test Group Deployment

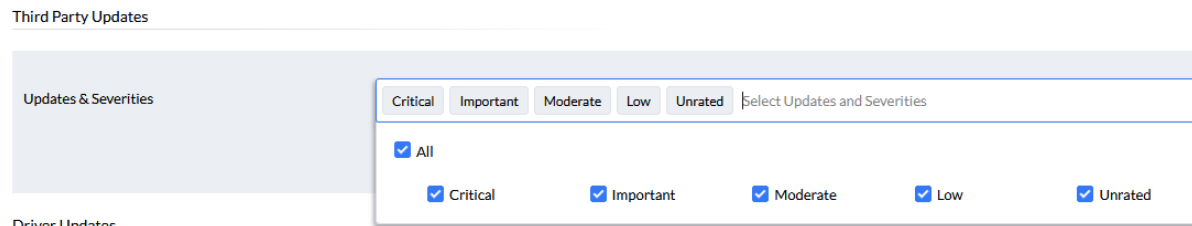


In den „Deployment Options“ wähle ich bei Microsoft Updates wie folgt:



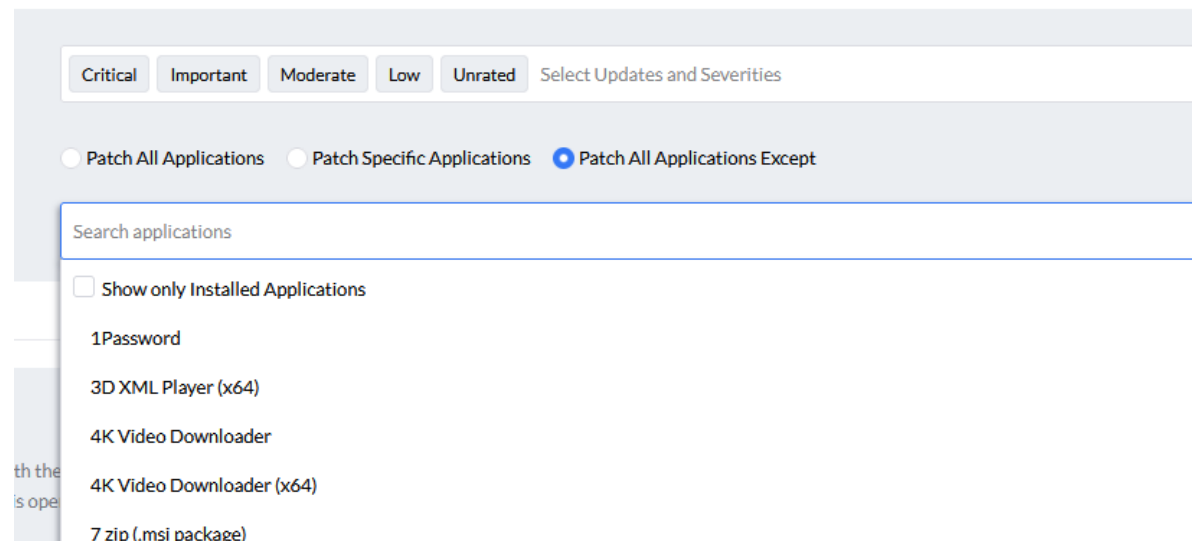
Service Packs, Feature Packs und der Gleichen teste ich weiterhin manuell oder erstelle eine spezielle Testgruppe hierfür.

Bei Third Party Updates definiere ich alles.



Driver Updates

Wenn ich aus den Auswahldialogen gehe, kann ich spezielle Anwendungen bei Microsoft Updates und Third Party Updates auch vom Patchen ausschließen. Z.B. spezielle VPN Lösungen, Workspace Applikationen, SQL, Java, etc.



Treiber Updates teste ich persönlich manuell oder teste diese über eine spezielle Gruppe an Geräten in einem weiteren Automatismus.

Deployment Criteria, damit kann ich eine Verzögerung definieren, also einen Zeitpunkt ab dem Patch-Release, ab welchem das Patch ausgerollt werden soll.

Ich belasse das auf 0 Tage, da ich in eine Testgruppe verteile und ich eine Freigabe des Patches im späteren Verlauf, nicht noch länger hinausschieben möchte.

Ja – auch wenn es bedeutet, dass ein System u.U. einmal ausfallen kann! Dann setze ich das System einfach neu auf, kompromittiere damit aber nicht meine Gesamtsicherheit.

**Deployment Criteria**

Only patches that have not been marked as Approved or Declined will be deployed to the Test Group

Deploy patches after  Days from vendor release

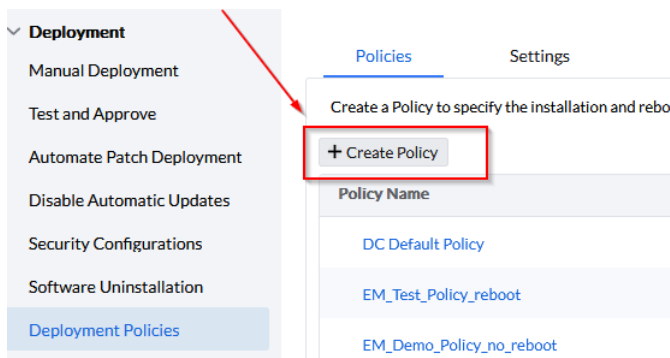
Im nächsten Schritt wähle ich die Deployment Policy – also das „Wann und Wie“ der Verteilung.

Deployment Settings

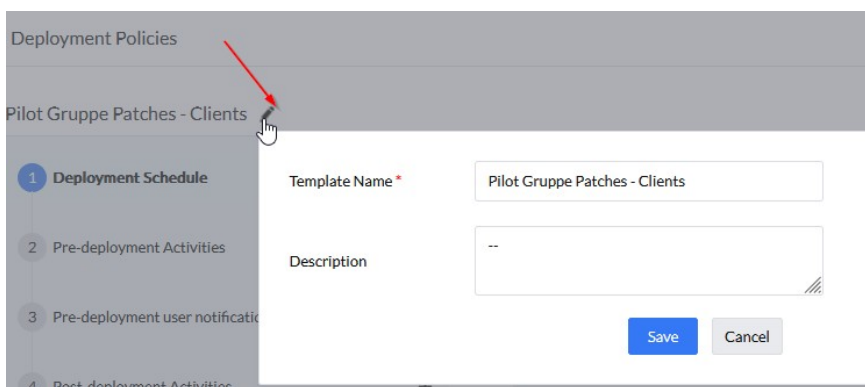
Apply Deployment Policy \* Pilot Gruppe Patches - Clients [View Details](#)

Hier erstelle ich eine eigene Policy, um das Verteilungsfenster auch unabhängig anpassen zu können.

Also rauf auf „Create/Modify/Save As Policy“, was mich in ein neues Tab, zu den Deployment Policies führt. Hier auf „Create Policy“ gehen.



Aussagekräftigen Namen geben.



Im Schritt 1 gebe ich das Zeitfenster an. Ich bevorzuge hier täglich wochentags (wenn die IT am Wochenende besetzt ist, dann auch Samstag und Sonntag), sowie zwei Zeitfenster, um auch auf flexiblere Arbeitszeiten zu reagieren. Man könnte jetzt fragen warum nicht gleich 24/7 patchen oder von 07:00 – 21:00 Uhr. Mein Gedanke bei einer festen Patch-Zeit ist, das ich auch die freundlichen Key-User nicht den ganzen Tag stressen möchte. Ich kann so den freundlichen Helfern eine fest Zeit geben in der die Patches installiert werden und sie können sich darauf einstellen. Die Mittagszeit bietet sich dazu auch am besten an.

Specify when patches/packages should be deployed to the client machines.

Week Split type:  Regular Split  Based on Patch Tuesday (Tue to next Mon)

Schedule Name	Preferred day(s)	Deployment Window	Actions
Schedule1	First, Second, Third, Fourth, Last - Mon, Tue, Wed, Thu, Fri	11:00 to 14:30	⋮
Schedule2	First, Second, Third, Fourth, Last - Mon, Tue, Wed, Thu, Fri	17:00 to 21:00	⋮

Patches lasse ich bereits Cashen wenn diese bereitstehen und verteilt werden soll zu jeder Zeit innerhalb des definierten Zeitfensters. So spare ich Bandbreite und Zeit.

Download patches from server to agent:  Only during Deployment Window  Any time agent contacts the server  
The agent will not wait for the deployment window to download the patch binaries. The patch binaries will be downloaded in the deployment window.

Initiate Deployment at:  System Startup  Refresh Cycle  Either of these whichever happens earlier  
Deployment happens either during the System startup or the Refresh cycle within the Deployment Window as chosen above

Nach „Save & Continue“ kann ich Pre-deployment Activities definieren. Wake-on-Lan sollte ich eine Verteilung nachts durchführen wollen, Pre-Deployment reboot oder ein Custom Skript laufen lassen. In meinem Beispiel definiere ich nichts und gehe gleich auf „Save & Continue“.

Deployment Schedule

Pre-deployment Activities:

Pre-deployment user notification

Post-deployment Activities

Schritt 3, bietet mir die Möglichkeit, die User vor der Verteilung zu benachrichtigen und ihnen ein Überspringen zu ermöglichen. Auch hier definiere ich nichts und gehe mit „Save & Continue“ weiter. Wenn ich ein Überspringen der Installation erlauben würde, kompromittiere ich meine Sicherheit. Außerdem werden die User dann wieder mit einem Dialog genervt. Wir sind aus dem digitalen Mittelalter heraus, Patches laufen zu 95% ohne eine Unterbrechung im Hintergrund.

Pre-deployment user notification

Notify user(s) about Deployment (Windows and Linux only)

Deployment Message Title \*

Notification Message during Deployment \*

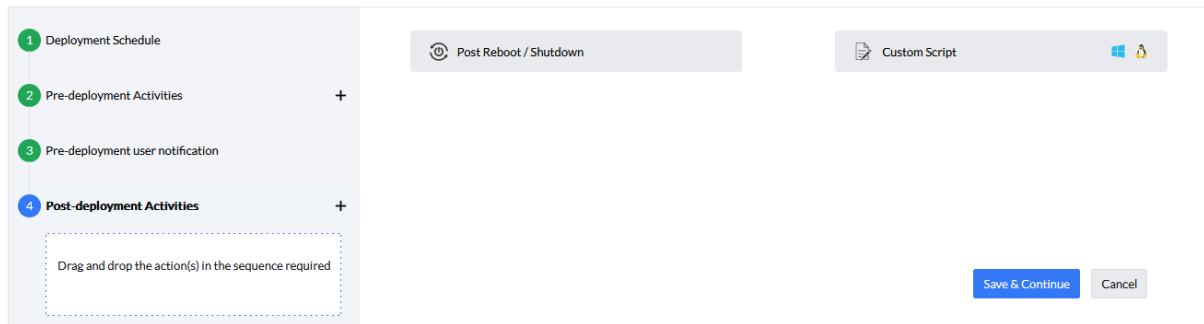
Notification timeout \*  minute(s)

Allow Users to Skip Deployment

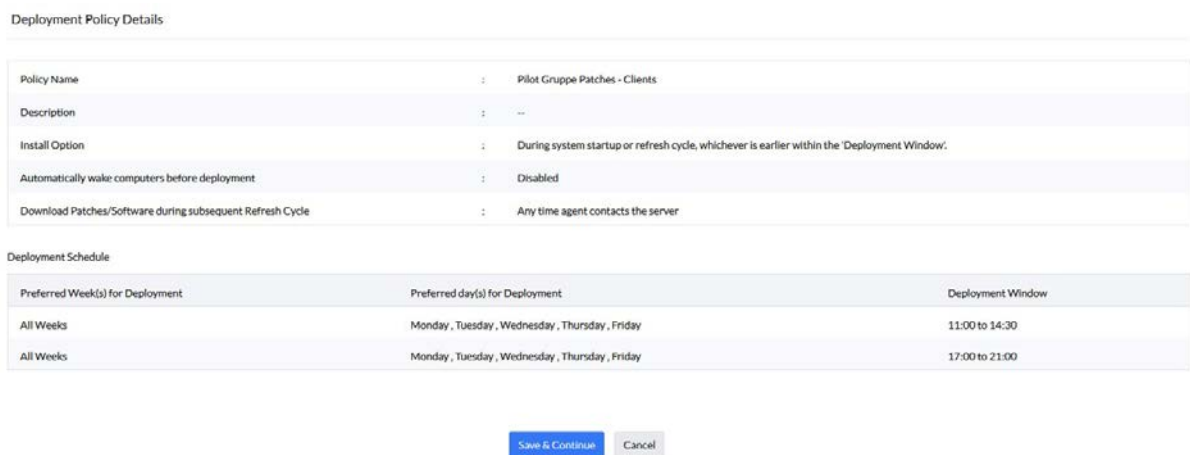
Show deployment progress on the client systems



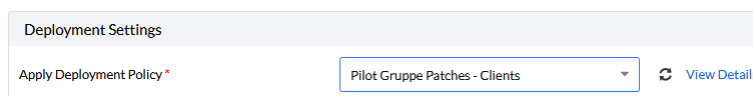
Schritt 4 und damit der Letzte bei der Definition der Deployment Policy, bietet mir die Möglichkeit einen Reboot oder Shutdown nach dem Deployment zu initiieren. Auch hiervon mache ich für den Fall der Key-User Pilotverteilung, keinen Gebrauch und gehe auf weiter mit „Save & Continue“. Bei Feature und Server Packs und dergleichen, erstelle ich eine manuelle Verteilung, welche dann auch mit einem Hinweis auf einen Reboot versehen ist. Ich halte diese Art der Patches aus dem täglichen automatischen Testverfahren heraus.



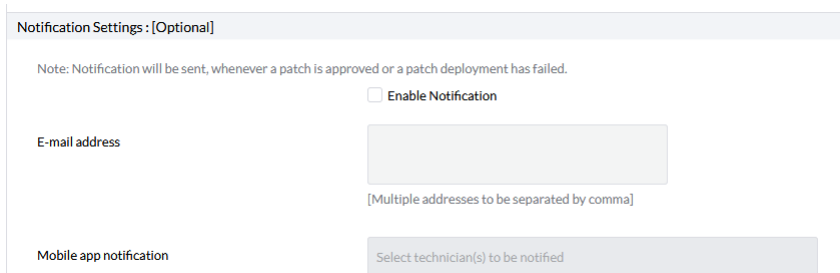
Nun erhalte ich noch einmal eine Übersicht und bestätige diese wieder mit „Save & Continue“.



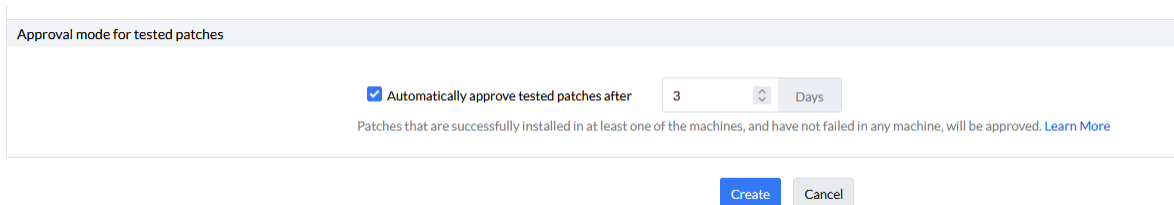
Zurück im Tab mit der Testgruppe welche wir erstellen, wähle ich die Policy aus.



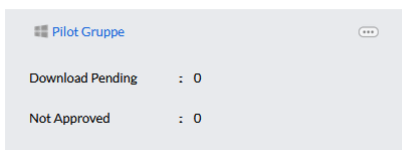
Bleibt die Frage nach Benachrichtigungen im nächsten Schritt. Ich persönlich halte von der E-Mail Flut nicht viel. Diese landen in einem Ordner, welcher irgendwann gelöscht wird. Ich baue auf die Dashboards, welche mir übersichtlich zu jeder Zeit einen Status geben. Dort kann ich ggfs. den fehlerhaften Patches nachgehen.



Letzter Schritt ist der Haken beim Approval Mode. Hier definiere ich drei Tage. **Mehr nicht** da ich ein weiteres Rollout der Patches nicht verzögern will. In den drei Tagen bis zur automatischen Freigabe der erfolgreich installierten Patches kann ich entweder über meine eigenen Programmaufrufe oder durch die Key-User in der Testgruppe eventuell ungewünschtes Programmverhalten identifizieren und diese Patches dann speziell ablehnen. Ein Patch sollte bei einer solchen Lösung nur dann automatisch freigegeben werden, wenn auf allen beteiligten Rechnern die Installation erfolgreich war. Bei EndpointCentral ist das gegeben. Wenn z.B. nur auf einem von 20 oder auch 100 Rechnern ein Patch mit einem Fehler antwortet, dann wird das Patch nicht freigegeben.



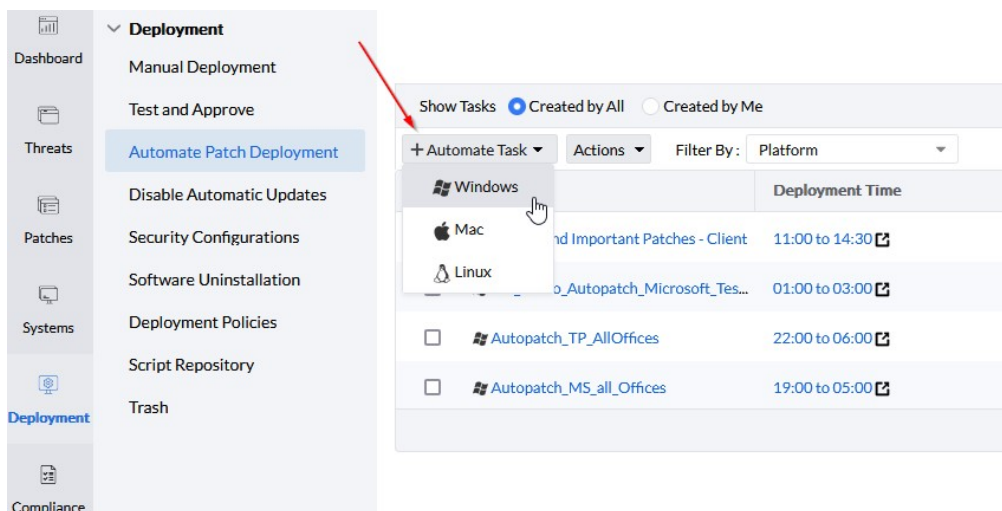
Ein Klick auf „Create“ erstellt mir dann meine Testgruppe.



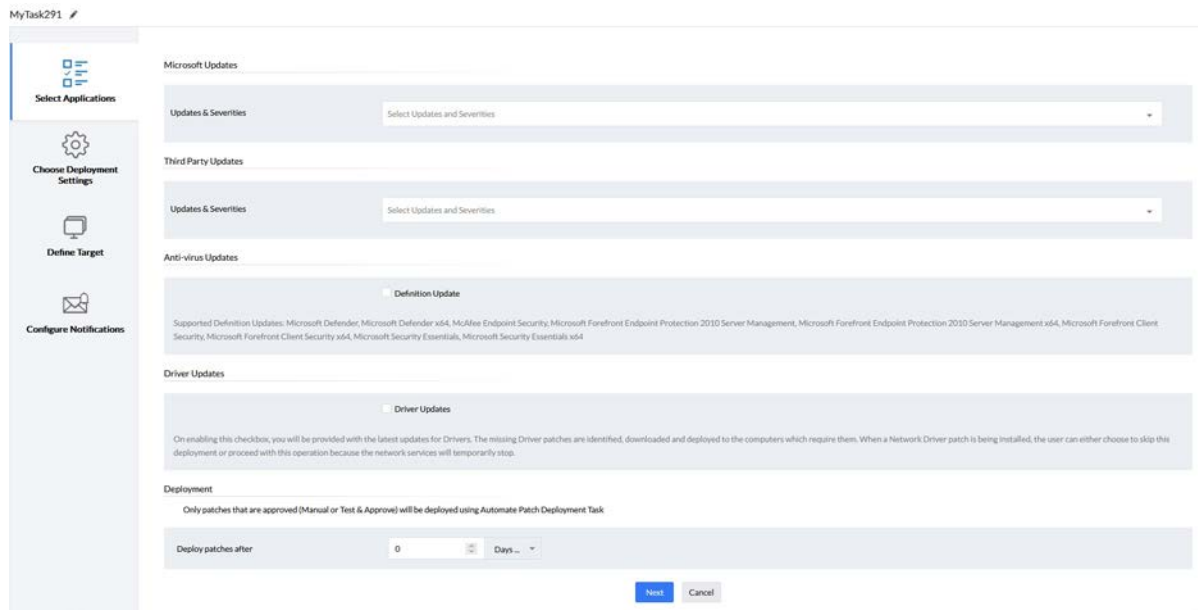
## 2.2 Automatische Verteilungen einrichten:

Nun, da ich ein Test und Freigabeverfahren eingerichtet habe, erstelle ich das erste automatische Deployment für die restlichen Endpoints im Unternehmen. Ich beginne mit den Critical and Important Patches.

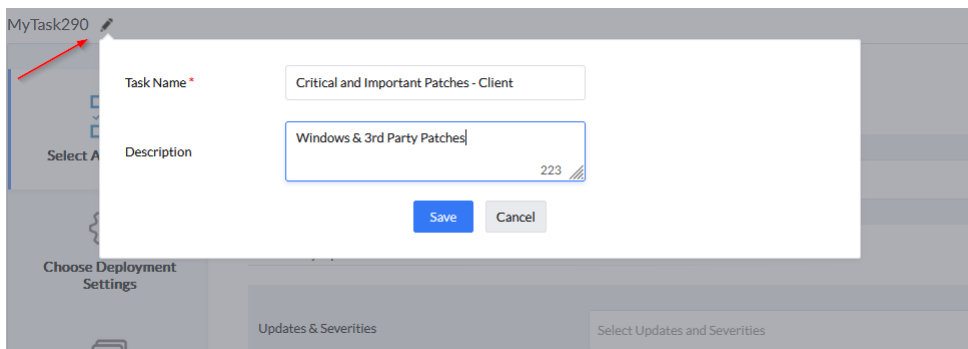
Hierzu in „Automate Patch Deployment“ wechseln und „Automate Task/Windows“.



Der Dialog ist ähnlich aufgebaut wie der aus der Testgruppe.



Wir geben dem Kind einen Namen und eine Beschreibung.



Bei den Microsoft und Third Party Patches wähle ich jeweils die Critical und Important aus und definiere eventuelle Ausnahmen für Software, welche nicht automatisch gepatcht werden soll.

Anti-Virus Updates überlasse ich der Virensoftware. Meiner Überzeugung nach sollte dies das Einzige sein, welches die Clients von selbst herunterladen können dürfen. So kann ich sichergehen, das Virenpatterns aktuell sind – auch in dem Falle der Agent auf dem System nicht mit dem EndpointCentral-Server kommunizieren kann und damit keine Updates gepusht werden.

Treiber Updates verteile ich persönlich manuell oder in einem separaten Automatismus.

Letzter Punkt auf dieser Seite ist der Zeitpunkt, wann das Patch verteilt werden soll. Da ich eine Verteilung nicht weiter verzögern will – wir haben bereits drei Tage aus dem Testverfahren – gebe ich hier null Tage ab Freigabe an. Außerdem kann ich auf diese Weise auch gerade erschienene High Critical Patches manuell testen und Freigeben. Durch meine manuelle Freigabe werden diese dann im automatischen Deployment sofort berücksichtigt. Ich muss hierfür dann keine separate Konfiguration erstellen – wieder Zeit gespart.

Microsoft Updates

Updates & Severities Security Updates [ Critical,Important ] [Select Updates and Severities](#)

Patch All Applications  Patch Specific Applications  Patch All Applications Except

---

Third Party Updates

Updates & Severities Critical Important [Select Updates and Severities](#)

Patch All Applications  Patch Specific Applications  Patch All Applications Except

---

Anti-virus Updates

Definition Update

Supported Definition Updates: Microsoft Defender, Microsoft Defender x64, McAfee Endpoint Security, Microsoft Forefront Endpoint Protection 2010 Server Management, Microsoft Forefront Security, Microsoft Forefront Client Security x64, Microsoft Security Essentials, Microsoft Security Essentials x64

---

Driver Updates

Driver Updates

On enabling this checkbox, you will be provided with the latest updates for Drivers. The missing Driver patches are identified, downloaded and deployed to the computers which require it or proceed with this operation because the network services will temporarily stop.

---

Deployment

Only patches that are approved (Manual or Test & Approve) will be deployed using [Advanced Deployment Task](#)

Deploy patches after  [Days ...](#)

[Days from release](#) [Days from approval](#)

[Next](#) [Cancel](#)

Weiter mit Next und ich gelange auf die Deployment Settings. Hier wieder über Create eine Neue erstellen. Keine Angst ... wir können die Deployment Policy, welche wir zuvor für die Pilot Gruppe erstellt haben, duplizieren. Hierzu einfach auf den Action Button der „Pilot Gruppe Patches – Clients“ Policy gehen und „Save As New“ klicken.

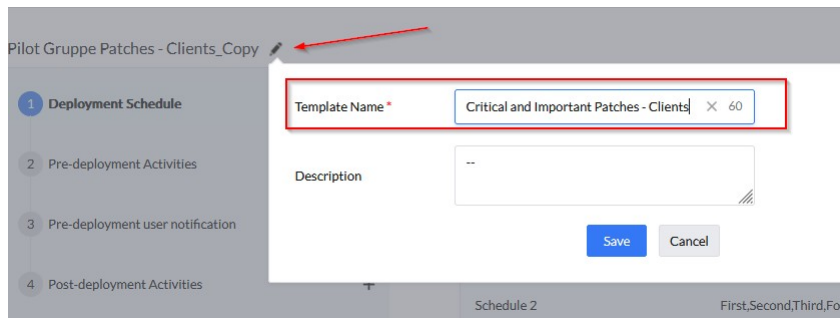
Deployment Option  Deploy  Publish to Self Service Portal (SSP)

Apply Deployment Policy Critical und Important Patches - Client [View Details](#) [Create](#)

Publish to Self Service Portal (SSP)  No

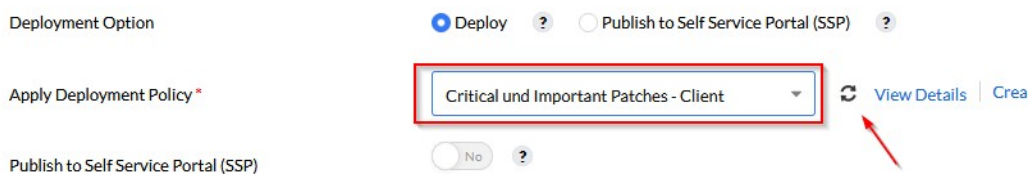
Pilot Gruppe Patches - Clients	admininelli	Jan 8, 2024 03:07 PM	admininelli	Delete	...
				Save As New	1

Wieder umbenennen und speichern.



Anschließend jeden Dialog mit „Safe & Continue“ abspeichern. Wir behalten alle Zeiten und Einstellungen bei.

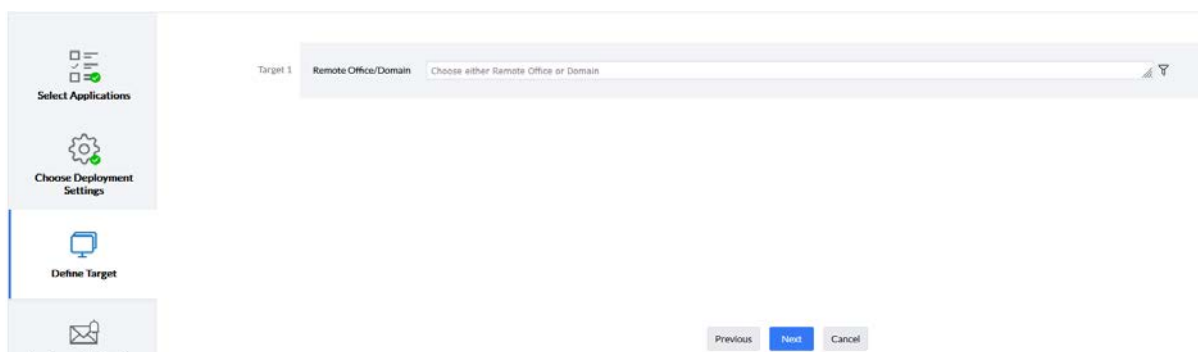
Zurück in das Tab mit der Automatischen Deployment Policy, können wir nach einem Klick auf den Refresh Button die eben erstellte Policy auswählen.



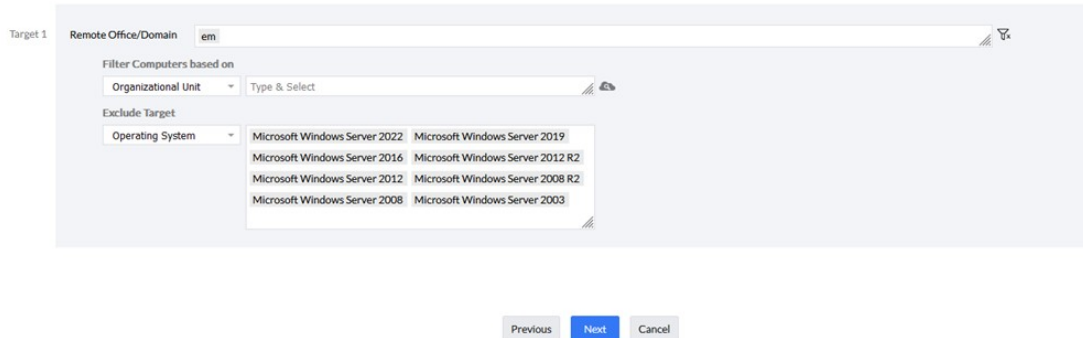
Patches können auch im Self-Service-Portal bereitgestellt werden. In meinem Fall belasse ich es beim normalen Deployment. Das Self-Service-Portal bietet sich vor allem dann an, wenn ich Feature oder Servicepacks verteilen möchte. Auch bei OS-Upgrades nutze ich das SB Portal und gebe den Mitarbeitern so die Möglichkeit ein zeitintensiveres Patchen, selbst zu planen.

Selbstverständlich gebe ich eine Deadline, zu welcher das Patchen dann erzwungen wird. Mir ist es wichtig, die User mit einzubinden.

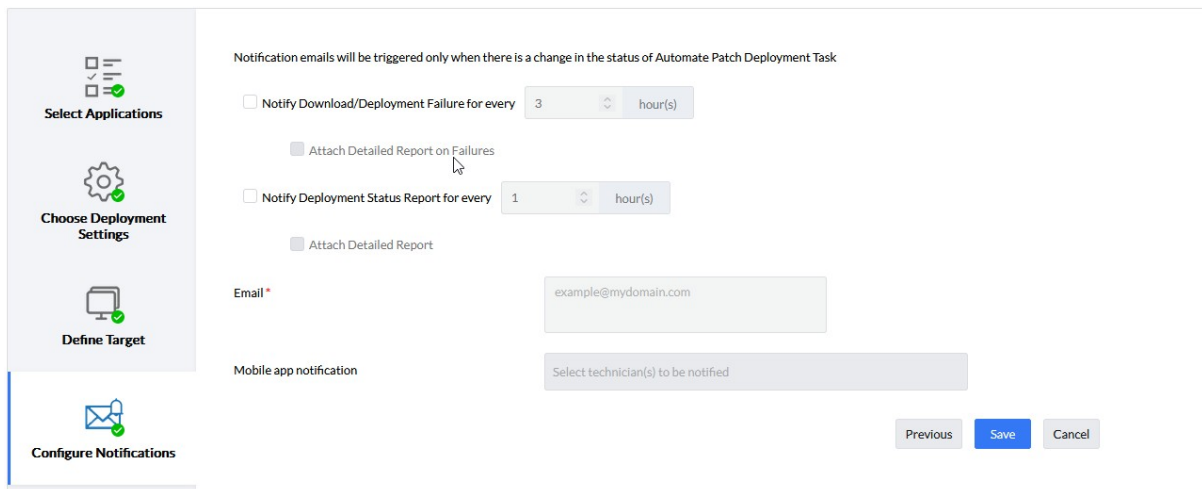
Über „Next“ gelange ich in die Ziele Definition.



Über das Targeting wähle ich ein Remote Office oder die Domäne, filtere nach den Systemen und exkludiere bestimmte Gruppen wie z.B. die Server Betriebssysteme oder Custom Groups, in welchen ich Systeme zusammengefasst habe, welche nichts **automatisiert** erhalten dürfen.



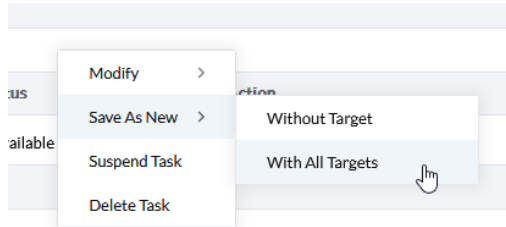
Beim Klick auf „Next“ gelange ich in den letzten Dialog mit den Benachrichtigungen. Auch hier sind E-Mail und Mobile-App Benachrichtigungen möglich. Ich selbst belasse es auch hier bei den Dash-Boards und klicke auf „Save“ ohne weitere Auswahl.



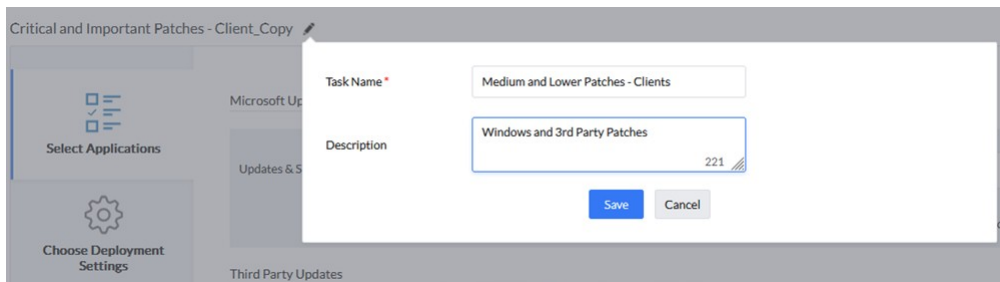
Trommelwirbel bitte ... fertig ist unsere erste von zwei automatischen Verteilungen.

Show Tasks <input type="radio"/> Created by All <input checked="" type="radio"/> Created by Me					
+ Automate Task <span style="float:right">Actions</span> Filter By: Platform					
<input type="checkbox"/>	Name	Deployment Time	Created Time	Current Status	Action
<input type="checkbox"/>	Critical and Important Patches - Client	11:00 to 14:30	Jan 8, 2024 03:54 PM	No targets available	...

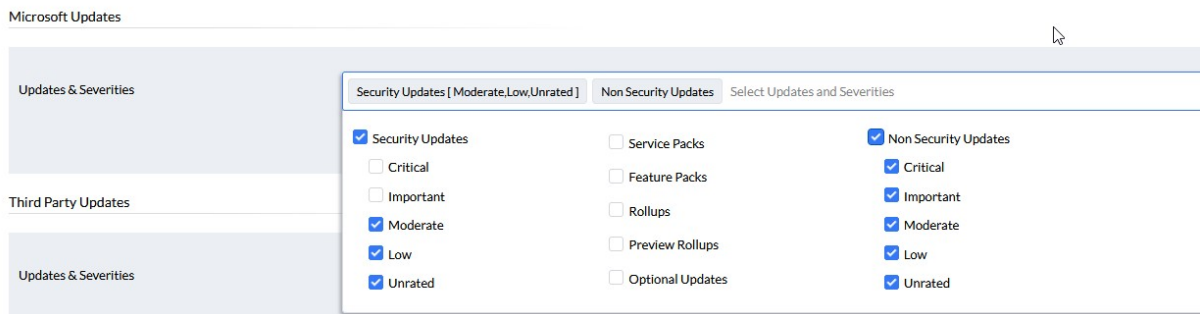
Nun dupliziere ich diese über den Action Button mit allen Zielen um auch die restlichen Patches von Medium abwärts, zu verteilen.



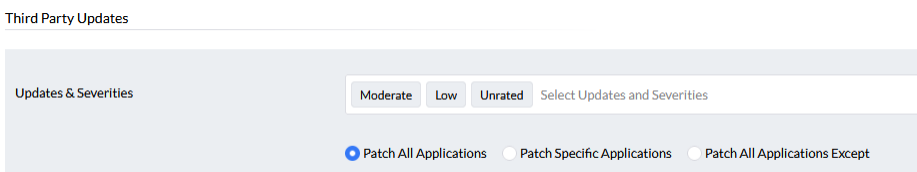
Kurz umbenennen.



Patches auswählen für Microsoft Updates. Optionale Updates verteile ich wie die Service und Feature Packs separat. Entweder in einem speziellen Automatismus oder über eine manuelle Konfiguration. Hier könnte sich ein monatlicher Zyklus anbieten.

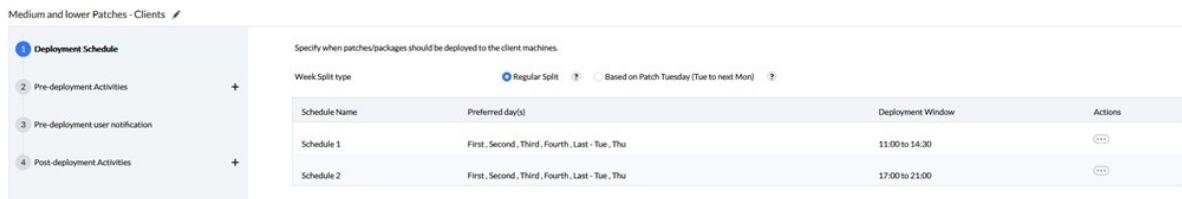


Gleiches gilt für die Third Party Patches.

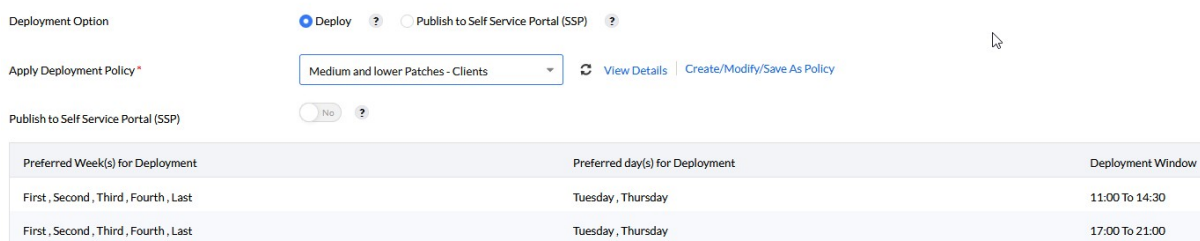


Nach dem Klick auf Next gelangen wir zu den Deployment Settings. Hier erstellen wir wieder eine Neue, da ich diese zum Zeitpunkt dieses Buches nur zwei Mal pro Woche verteile. Also wieder auf „Create/Modify/Save As Policy“ gehen.

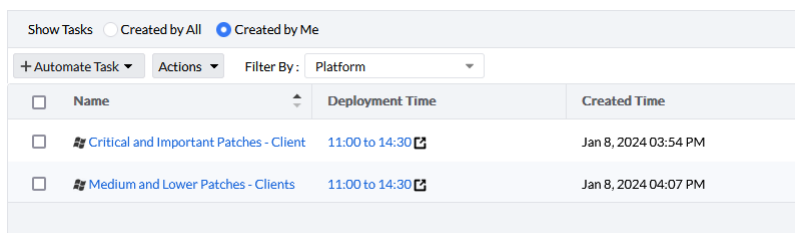
Ich dupliziere die Critical and Important Patches – Client Policy, benenne diese um und definiere zu denselben Uhrzeiten lediglich Dienstag und Donnerstag als Verteilungstage. Der Rest bleibt wie gehabt und ich klicke durch alle Dialoge mit „Save & Continue“.



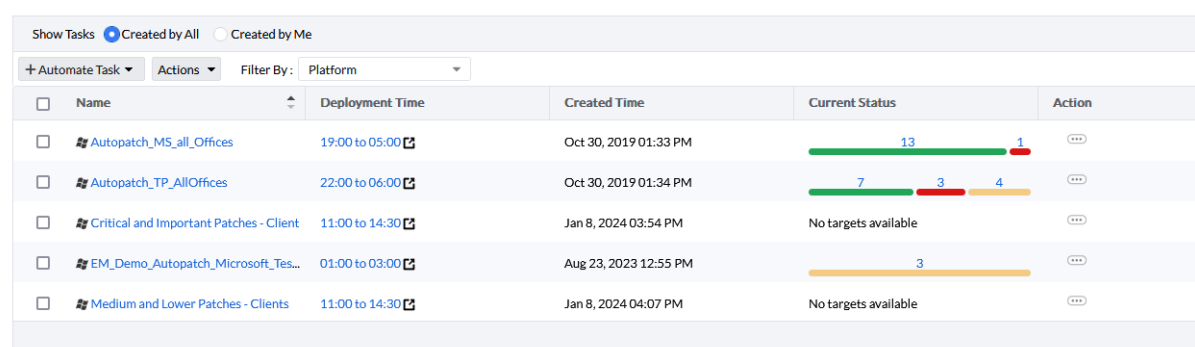
Zurück in unseren Automatismus Tab kann ich nach einem Refresh die neue Policy auswählen und bestätige alle weiteren Dialoge ohne Änderungen.



Im Anschluss habe ich meine zwei Automatismen, welche mir alle freigegebenen Patches im Unternehmen ausrollen.



Über die Statusanzeigen kann ich den Verlauf kontrollieren und ggfs. den fehlerhaften Verteilungen nachgehen.





Noch ein Tipp! Wenn ich in Kundenumgebungen komme, in welchen es bisher die Speedy Gonzales Turnschuh-Administration gab, sehe ich mir zuerst die Menge an fehlenden Patches in der Übersicht an. Sind es zu viele, erstelle ich zwar beide Automatismen wie oben beschrieben, suspendiere jedoch die für Medium and Lower Patches. Ich lasse die ersten zwei Wochen **allein** die Critical and Important Patches ausrollen. Auf diese Art werden die Systeme und mein Netzwerk nicht überstrapaziert. Ich lasse sozusagen etwas Druck ab.

## 2.3 Überprüfung weiterer Schwachstellen im Thread Management:

Es gibt neben der normalen Patches auch eine Menge an weiteren Schwachstellen, um welche ich mich kümmern darf. Misskonfigurationen, Zero-Day Schwachstellen, offene Ports, Web-Server Misskonfigurationen, etc.

In der EndpointCentral Security Edition sowie im VulnerabilityManager Plus, ist hierzu ein zusätzlicher Bereich „Threats“ geschaffen worden.

Hier finde ich noch einmal in Kategorien einteilbar eine Übersicht über meine noch offenen Schwachstellen inklusive ausführlicher Beschreibung und CVE Scores. Kann hier auch Ausnahmen definieren und/oder die Schwachstellen beheben.

The screenshot shows the 'Threats' interface. On the left, a sidebar lists categories like 'Software Vulnerabilities', 'Detected CVEs', 'Zero-day Vulnerabilities', etc. The main area shows a search bar and a list of vulnerabilities. One entry is expanded to show details for PostgreSQL, including CVE IDs (CVE-2020-25696, CVE-2020-25695, CVE-2020-25694), CVSS 3.0 score (9.8), and affected systems (EM-SRV-EXC01).

Es gibt auch einen Bereich, welcher Zero-Day Schwachstellen aufzeigt, und ich kann entsprechend handeln.

The screenshot shows the 'Zero-day Vulnerabilities' section. It features a table with columns for 'Threats', 'Threat Category', 'Affected Systems', and 'Action'. The table lists several vulnerabilities, including CVE-2023-7024 in Google Chrome and CVE-2022-41082 in Microsoft Exchange Server. The 'Action' column shows options like 'Fix' or 'No fix available'.

Threats	Threat Category	Affected Systems	Action
Vulnerabilities CVE-2023-7024 are fixed in Google Chrome (120.0.6099.129, 120.0.6099.130)	CVE-2023-7024	1	Fix
Vulnerabilities CVE-2023-7024 are fixed in Google Chrome (x64) (120.0.6099.129, 120.0.6099.130)	CVE-2023-7024	1	Fix
Windows Media Remote Code Execution Vulnerability for Windows Server 2016 for x64-based Systems	CVE-2023-20588	1	Fix
Vulnerabilities CVE-2023-3079 are fixed in Google Chrome (114.0.5735.110)	CVE-2023-3079	2	Fix
Vulnerabilities CVE-2023-2133, CVE-2023-2134, CVE-2023-2135, CVE-2023-2136, CVE-2023-2137, and CVE-2023-2138	CVE-2023-2136	1	Fix
Microsoft Exchange Server: Remote Code Execution Vulnerability (CVE-2022-41082)	CVE-2022-41082	1	No fix available
Microsoft Exchange Server: Elevation of Privilege Vulnerability (CVE-2022-41040)	CVE-2022-41040	1	No fix available
GhostCat: Vulnerabilities CVE-2020-1938, CVE-2020-1935, CVE-2019-17569 are fixed in 11 February 2020	GHOST_CAT_VULNERABILITY	1	No fix available

Im Bereich „System Misconfigurations“ schließe ich, was mir möglich ist oder definiere Ausnahmen. Natürlich kann vieles davon über die Active Directory konfiguriert werden. Ich kann die Ergebnisse dieser Überprüfung nutzen und ggfs. direkt in der AD konfigurieren.

The screenshot shows the 'System Misconfigurations' section in dpoin Central. It displays a table with columns: Misconfiguration, Category, Affected Systems, Reboot Required, and Action.

Misconfiguration	Category	Affected Systems	Reboot Required	Action
Administrative Shares enabled	Share Permission Management	13	Not Required	No fix available
Geolocation is enabled to track users physical...	Chrome Security Hardening	10	Not Required	Deploy Secure Configuration
TLSv1.1 protocol is enabled	SSL and TLS Security	13	Required	No fix available
The Server Message Block (SMB) v1 protocol...	Legacy Protocols	1	Not Required	Deploy Secure Configuration
Windows Credential Guard has been found di...	OS Security Hardening	13	Required	No fix available
Insecure RC4 cipher algorithms are not disabl...	SSL and TLS Security	2	Required	No fix available
Administrator accounts are enumerated dur...	Logon Security	13	Not Required	Deploy Secure Configuration
User rights granted to everyone group	Account Privilege Management	13	Not Required	No fix available

The screenshot shows the detailed view for 'Chrome Security Hardening' with 15 items. It lists various security settings and their status, along with the number of affected systems and the required action.

Category	Item	Status	Action
Info	Geolocation is enabled to track users physical location	Not Configured	Deploy Secure Configuration
Moderate	Firewall traversal from remote host is not disabled	Not Configured	Deploy Secure Configuration
Moderate	Automatic update of Chrome browser is not enabled	Not Configured	Deploy Secure Configuration
Moderate	Chrome minimum TLS/SSL connection must be configured to TLS 1.2	Not Configured	Deploy Secure Configuration
Moderate	Websites must be prevented from accessing USB devices	Not Configured	Deploy Secure Configuration
Moderate	Anonymized URL-keyed data collection by Google must be disabled	Not Configured	Deploy Secure Configuration
Moderate	Download restrictions must be configured	Not Configured	Deploy Secure Configuration
Moderate	Chrome background apps running continuously	Not Configured	Deploy Secure Configuration
Moderate	Ensure browser history is saved	Not Configured	Deploy Secure Configuration
Moderate	Importing of saved passwords is not disabled	Not Configured	Deploy Secure Configuration
Moderate	Anonymous browser usage and crash-related data collection by Google is not disabled	Not Configured	Deploy Secure Configuration
Moderate	Chrome password manager is not disabled	Not Configured	Deploy Secure Configuration
Moderate	Cloud print sharing via Chrome is not disabled	Not Configured	Deploy Secure Configuration
Moderate	Pop-ups must be blocked	Not Configured	Deploy Secure Configuration
Low	Third party cookies must be blocked	Not Configured	Deploy Secure Configuration

Dann sehe ich nach ob es abgelaufene Software unter „High Risk Software“ gibt bzw. wann vorhandene Software ihr Laufzeitende erreicht. Ein Thema, welches oft vernachlässigt wird und „urplötzlich“ auftaucht.

The screenshot shows the 'High Risk Software' section in dpoin Central. It displays a table with columns: Software Name, Vendor, Expiry Date, Days to Expire, and Affected Systems.

Software Name	Vendor	Expiry Date	Days to Expire	Affected Systems
Microsoft Visual C 2008 Redistributable (x86)	Microsoft	Apr 9, 2013	Expired	5
Microsoft Visual C++ 2010 Redistributable (x64)	Microsoft	Jul 10, 2015	Expired	1
Microsoft Visual C++ 2010 Redistributable (x64)	Microsoft	Jul 10, 2015	Expired	4
Microsoft Visual C++ 2010 Redistributable (x86)	Microsoft	Jul 10, 2015	Expired	1
Microsoft Visual C++ 2010 Redistributable (x86)	Microsoft	Jul 10, 2015	Expired	4
Microsoft Visual C++ 2008 Redistributable (x64)	Microsoft	Apr 10, 2018	Expired	4
Windows Server 2016 Datacenter Edition (x64)	Microsoft	Jan 10, 2022	Expired	1
Microsoft SQL Server 2017 Developer Edition (x...	Microsoft	Oct 11, 2022	Expired	1
Windows Server 2012 R2 Datacenter Edition (x64)	Microsoft	Oct 9, 2023	Expired	1
PostgreSQL (Less than version 12)	PostgreSQL	Nov 9, 2023	Expired	7
Tomcat 8.5	Tomcat	Mar 31, 2024	81 days left to expire	4
Python 3.9	Python Software Foundation	Oct 31, 2025	660 days left to expire	1
PostgreSQL 14	PostgreSQL Global Development Group	Nov 12, 2026	1,037 days left to expire	4
.NET Framework 3.5 (x64)	Microsoft	Jan 10, 2029	1,827 days left to expire	5

Weitere Schwachstellen betreffen die Webserver. In diesem Bereich bekomme ich eine Übersicht mit Lösungsvorschlägen, mit deren Hilfe ich Schwachstellen auf den Webservern beheben kann.

The screenshot shows a security dashboard with a sidebar on the left containing categories like 'Zero-day Vulnerabilities', 'System Misconfigurations', 'High Risk Software', 'Web Server Misconfiguration', 'Port Audit', and 'Manage Exceptions'. The main content area is titled 'SSL' and shows 26 items. A detailed view of the SSL section lists 15 items with their severity levels:

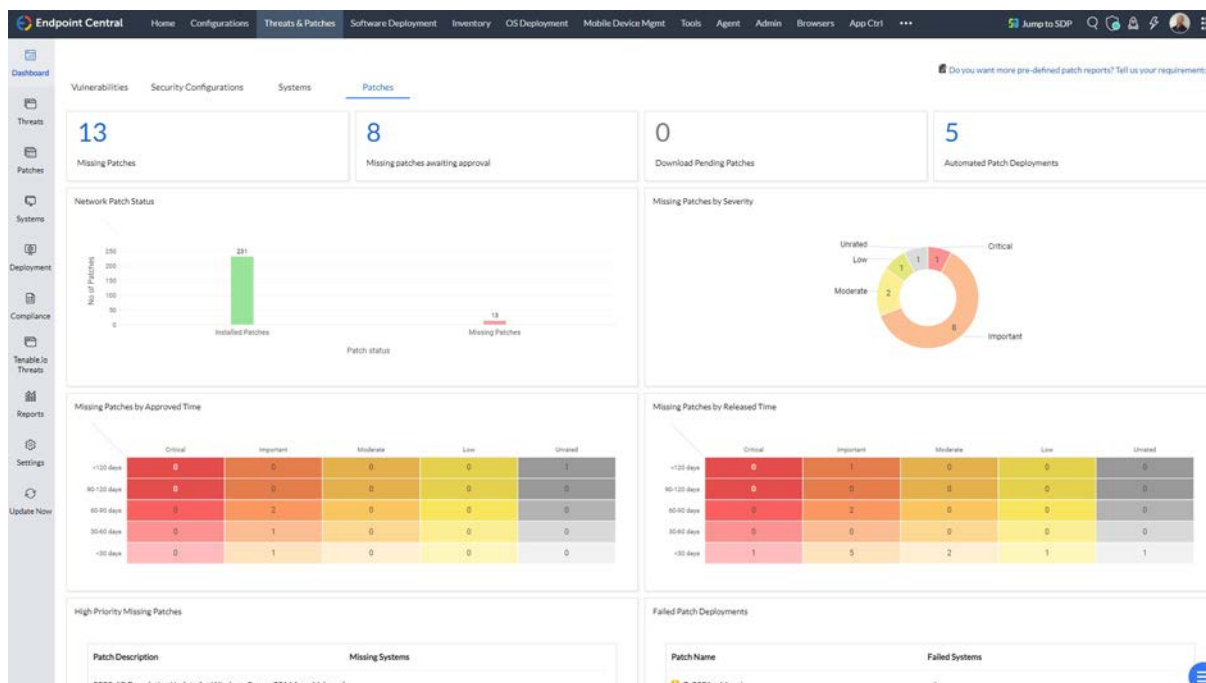
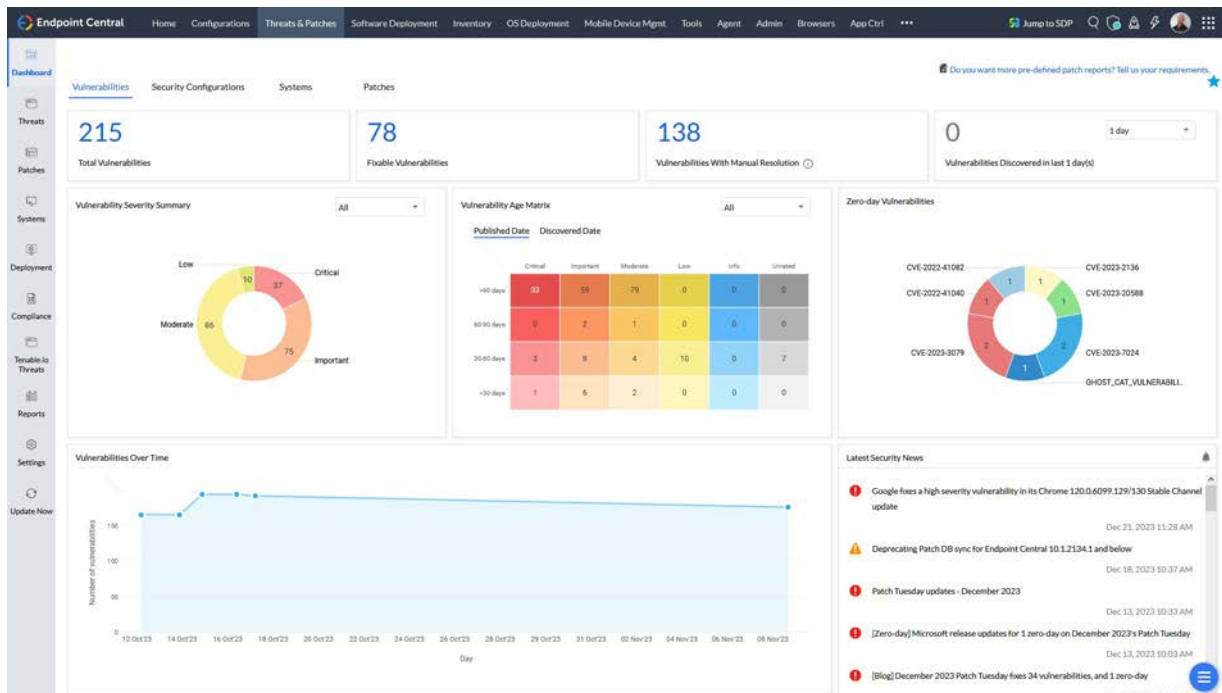
- Info: Ensure SSLEnabled is set to True for Sensitive Connectors
- Info: Ensure scheme is set accurately
- Info: TLS 1.1 is enabled (IIS)
- Critical: Ensure HSTS Header is set
- Critical: Ensure TLS 1.0 is disabled (IIS)
- Critical: HTTPS is not configured
- Critical: Ensure RC4 Cipher Suites are disabled (IIS)
- Critical: There are no intermediate certificates installed due to which the SSL chain is incomplete
- Critical: Tomcat supports TLSV1.0 protocol
- Important: Tomcat server is not restricted from using RC4 algorithm
- Important: DES and 3DES cipher algorithms that are prone to Birthday attacks are not disabled in Tomcat server
- Important: Weak TLS/SSL ciphers are not disabled in Tomcat server
- Important: Tomcat supports MEDIUM, LOW or EXPORT ciphers
- Important: Tomcat server uses default cipher suites
- Important: Insecure MD5 hashing algorithm is not disabled (IIS)
- Important: Nginx server supports MEDIUM, LOW or EXPORT ciphers

Und zum Schluss sehe ich mir an, welche Ports auf welchen Systemen offen sind.

Port Number	Port Type	Instances	Description
22	TCP	2	The Secure Shell (SSH) Protocol
25	TCP	1	Simple Mail Transfer
69	UDP	6	Trivial File Transfer
80	TCP	4	World Wide Web HTTP
81	TCP	2	NT Kernel & System; Apache HTTP Server
123	UDP	14	Network Time Protocol
135	TCP	14	DCE endpoint resolution
137	UDP	13	NETBIOS Name Service
138	UDP	13	NETBIOS Datagram Service
139	TCP	13	NETBIOS Session Service
161	UDP	14	SNMP
162	UDP	4	SNMPTRAP
443	TCP	7	http protocol over TLS/SSL
444	TCP	1	Simple Network Paging Protocol

## 2.4 Zeit für den Cappuccino – Dashboards prüfen:

Anstelle mich durch eine Flut an Benachrichtigungs- und Status E-Mails zu wühlen, gehe ich in die Dashboards. Das Threats & Patches Dashboard liefert mir alle nötigen Informationen auf einen Blick und ich wechsele durch das Dashboard in Bereiche welche meiner Aufmerksamkeit bedürfen. Ein übersichtliches Dashboard mit direkten Verlinkungen ist ein MUSS für mich.



Einmal eine saubere Patchstrategie definiert brauche ich mich nur noch um die noch fehlenden nicht freigegebenen Patches kümmern und ggfs. ein Troubleshooting, wenn auf einem System ein Patch nicht erfolgreich installiert werden konnte.

Zugegeben, es ist ein initialer Aufwand eine Strategie zu konfigurieren. Einmal eingerichtet jedoch, reduziert es meinen bisherigen manuellen Aufwand erheblich und bietet maximalen Schutz der Endpoints. Was mir an der Lösung von ManageEngine so gefällt ist, das ich übersichtlich und einfach Patchautomatismen anlegen kann – ohne Raketenwissenschaften studiert zu haben. Auch brauche ich mit guten Dashboards und Reports keine Glaskugel zu Rate zu ziehen und orakeln lassen wie sicher meine Clients sind.

Das ist meine Strategie beim Endpoint Patching. Durch diese schaffe ich mir freie Zeit, welche ich nutzen kann um die weiteren Schwachstellen wie USB Devices, Applikationen, Browser, Bitlocker, etc. anzugehen. Auch hierzu gibt es viele Lösungen auf dem Markt und natürlich auch in meiner bevorzugten Managementlösung EndpointCentral.

Gerade in einer Zeit in welcher ich es schwer habe neue Kollegen in der IT zu finden, benötige ich eine Lösung, welche mir Freiräume schafft. Mit EndpointCentral habe ich solch eine Lösung, bei welcher ich außerdem einen exzellenten Support von ManageEngine und dem deutschen Partner MicroNova AG, zur Verfügung habe.

Also worauf noch warten?

Viel Spaß und haltet Eure Endpoints Up-To-Date!

Jürgen Rinelli

### 3. About the Author:

MCITP, MCTS, MCP, MOS, Enterprise Administrator, Senior Software Consultant, SCCM-Spezialist, Autor, Coach, Reiki-Lehrer ...

Jürgen Rinelli wurde 1970 in Deutschland geboren. In seinem ereignisreichen und oft abenteuerlichen Leben hat er in vielen Ländern gelebt und gearbeitet. Ob als Geschäftsmann, Manager, Mechaniker, Trainer, Taucher oder IT-Experte, er findet immer einen Weg, seine Träume zu verfolgen.

