

PRTG – Citrix ADM Nutzen um ungespeicherte ADC Konfigurationen zu überwachen

Im einem der letzten Beiträge habe ich ein Powershell Modul vorgestellt, mit welchem man Daten vom Citrix ADM abgreifen kann.

Basierend auf diesem Modul habe ich bei uns einen PRTG Custom Sensor erstellt, welcher ungespeicherte Konfigurationen aus dem ADM ausliest:

Configuration Audit



Dazu müssen erst einmal die Parameter von PRTG sowie das Modul selbst eingelesen und eine Verbindung zur Appliance hergestellt werden:

```
# Get parameter from PRTG
param (
    [string]$server,
    [string]$domain,
    [string]$username,
    [string]$password
)

# Import ADM PS module
$CustomSensors="C:\Program Files (x86)\PRTG Network
Monitor\Custom Sensors\EXEXML\"
Import-Module $CustomSensors\PRTGCustomCitrixADM.psml

# Create the ADM session
```

```
$ADMHost = "https://" + $server
$ADMSession = Connect-ADM -ADMHost $ADMHost -CredUser
$username -CredPW $password
```

Damit dies funktioniert, muss der entsprechende Benutzer von PRTG auf dem ADM Leserechte haben. Weiter gehe ich nicht auf jede Zeile ein, dies kann im Script selbst nachgelesen werden. Hier beschreibe ich vor allem die wichtigsten Punkte, wie das Auslesen der Konfigurationsdaten aus dem ADM:

```
# Get the config status from the ADM
$DiffEvents = Invoke-ADMNitro -ADMSession $ADMSession -
OperationMethod GET -ResourceType ns_conf
# Create the variable only with the active events content
$DiffEvents2 = $DiffEvents | Select-Object ns_conf
```

Nun werden die Daten noch auf den String „Diff Exists“ geprüft und für die weitere Auswertung aufbereitet. In diesem Sensor gibt es nur einen guten oder schlechten Rückgabewert, weshalb der Block sehr kurz gehalten ist:

```
$Events = @()
ForEach ($Event in $DiffEvents2.ns_conf){

    If ($Event.diff_status -eq "Diff Exists"){
        # Filter out 'entityup' messages from critical state
        $RetState = $returnStateCritical

        $Events +=
[PSCustomObject]@{Severity=$Event.hostname;SourceIP=$Event.ns_
ip_address;SourceHost=$Event.hostname;State=[Int64]$RetState}
        $RetCritical = $RetCritical + 1
    }
}
```

Im restlichen Skript wurden die einzelnen Daten mit einem vorgegebenen Wert gegen geprüft und die entsprechenden PRTG Ausgaben definiert.

Das Script muss nun in den Custom Sensors Ordner von PRTG gelegt werden und der Sensor kann entsprechend eingerichtet werden:



Parameter gem. Script-Beschreibung, Securitykontext vom Gerät



Der Intervall wird ein wenig höher gestellt, damit bei aktiven Änderungen das Monitoring nicht sofort Alarm schlägt.

Das Resultat sieht dann wie folgt aus:



Viel Spass beim Nachbauen :-)

PRTGCustomCitrixADMConfigDrift.ps1_Herunterladen