

Continue



Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Note: All below commands need you to have access to Qradar through SSH (putty) OR direct screen and keyboard or KVM How to capture traffic/complete communications taking place on a particular interface of QRadar. Mostly or always the main interface of QRadar is named as eth0 and you have the option to change it if it is required (this will be discussed later) tcpdump -i eth0 How to capture data arriving to QRadar on particular interface plus using port number e.g. syslog port 514 so it will filter all other traffic and just dump what is coming through 514 tcpdump -i eth0 port 514 How to capture data arriving to QRadar box from a specific IP address tcpdump -n src host x.x.x.x How to combine the captured data arriving to QRadar with a particular IP and Port at the same time tcpdump -A -s 0 -j eth0 -n port 514 and host x.x.x.x How to get/capture a verbose traffic going or coming from a specified host IP tcpdump -n -A -s 0 src host x.x.x.x How capture everything coming to QRadar from a particular host on particular interface (I use this mostly before adding a log source to QRadar I just run the command and start working on the log source till i get the result from this command) tcpdump -npi eth0 src host x.x.x.x How to get the EPS of your box from the command line login to Qradar and cd var/log tail -f qradar.log | grep 'Events per second' How to generates a report on the dropped events/flows in the pipeline login to QRadar cd/opt/qradar/bin ./dumpMBeanSummary.sh How to get real time errors of QRadar tail -f /var/log/qradar.error How to change your QRadar IP/gateway/dns/email server IP ,.etc sittings this cannot be done through SSH connect a scree and keyboard directly to your box login with root credentials /opt/qradar/bin/qchange netsetup How to examine or get the current license of your box cat /opt/qradar/conf/license How to display the serial number of the system /opt/qradar/bin/getserial OR dmi decode | grep serial How to get all the info of your QRadar e.g. Appliance type, Core version of the system, Patch number, Is the QRM enabled, What's the IP address, Is the appliance you ran this command is a console, What's the kernel architecture, Information about CPU, Operating System and if this is HA host or not. /opt/qradar/bin/myver -v How to monitor/check disk size of QRadar and get the result into txt file specially when it is in distributed model or having different managed host if it is in stand alone you may use df -h if it has other managed hosts use /opt/qradar/support/all_servers.sh 'df -h' > /root/disksize.txt How to restart Qradar services This here would be the most important as well ignored aspect of Qradar operations. The right Order, in real-world I have seen whole bunch of horrible things happening when the right order or sequences for start/stopping services are not used. Those bad-evil things at worst-case scenario, has even lead to rebuilding of entire system. So for console the order is service tomcat stop service hostcontext stop service hostservices stop To start (in reverse) service hostservices start service hostcontext start service tomcat start Brief desc about these services:- Host Context - Monitors all QRadar components to ensure that each component is operating as expected. Tomcat - frontend web-server responsible for all UI interactions. Troubleshooting/Troubleshooting Checklistssh -vv/opt/qradar/support/deployment_info.sh -OHall hosts: /opt/qradar/support/all_servers.sh -Ck "/opt/qradar/bin/myver -tf"systemctl status hostcontextdf -This /opt/qradar/conf/hostcontext.NODOWNLOAD/opt/qradar/support/validate_deployment.sh md5sum locate host.token` ` bandwidth: grep "Replication download" /var/log/qradar.log | tailvar/log/setup/patches.logvar/log/qradar.error/store/backup/marathon■ Pretest:/opt/qradar/support/validate_deployment.sh/opt/qradar/support/all_servers.sh -Ck "/opt/qradar/bin/myver -tf"df -h /storetmp /var/log | tee diskchecks.txt/opt/qradar/support/apar/aqlValidatorum list all | grep -i AGENT-WINCOLLECTpsql -U qradar -c "select * from ale_component type"tail -f /var/log/qradar.log | grep -i 'incoming raw /opt/qradar/support/jmx.sh 7782 | grep Queries/opt/qradar/support/jmx.sh 7782 | grep Querieswatch -n1 /opt/qradar/support/jmx.sh -p 7787 -b"com.q1labs.sem:application=ecs-ec-ingress.ecs-ec-ingress,type=sources,name=Source Monitor""echo "COPY (select fg.name, sd.id, sd.hostname, sd.device name, st.device type name, sd.device description,(to timestamp(sd.timestamp last seen)/1000), sd.device enabled from sensor device sd left join sensor device type st on sd.device type peid=st.id left join fgroup link fl on trim(both ' ' from to_char(sd.id, '99999')))=fl.item id left join fgroup fg on fl.fgroup_id=fg.id where sd.device enabled='t' order by fg.name) TO STDOUT with CSV HEADER" | psql -U qradar -o /root/SIEMLogsources.csv qradar/opt/qradar/support/get_logs.sh -Spsql -U qradar -c "select i.id,i.name,i.status,i.task status,i.managed host_id, m.hostname from installed_application instance as i left join managed host as m on m.id = i.managed host_id;"Check /opt/qradar/conf/deployment.xml Support Tools 101 QRadar Support tools and commands for use in the administration and troubleshooting of a QRadar system. Ask about tools in our forums About Using this Page Support Tools 101 is intended for administrators and IT Professionals who are responsible for troubleshooting and working with QRadar Support to maintain their QRadar environment. This page contains scripts and commands used to gather information on appliances, troubleshoot specific features, and assist in technical resolutions. We advise administrators to not use tools if you are unfamiliar with their functionality or with a documented option flag. If you have a question regarding functionality, how-to questions, or if the tool does not work as designed, submit the question in our forums using the link at the top of the page. Support cases for these scripts are out of scope. Category Name Description and Examples System myver The script provides the current version, patch, and other system information for a QRadar system. /opt/qradar/bin/myver -v System deployment_info.sh This tool collects all information about all systems in the deployment, including disk space used, hardware, appliance type, and serial number within a CSV file. /opt/qradar/support/deployment_info.sh -OS Services validate_ecs_services.sh This tool can be used to check the connections to all managed hosts and verify the versions of ECS and ECS-Ingress services after an upgrade. /opt/qradar/support/validate_ecs_services.sh Services wait_for_start.sh The script monitors and displays the status of the hostcontext processes, whether they are running or stopped on a QRadar system. /opt/qradar/upgrade/util/setup/upgrade/wait_for_start.sh Troubleshooting all_servers.sh The all_servers.sh command is a powerful tool that can issue commands to all QRadar appliances within your deployment. /opt/qradar/support/all_servers.sh -h Administration changePasswd.sh The change password tools allows you to change the admin account password using the CLI in an incorrect password recovery scenario. /opt/qradar/support/changePasswd.sh -a Health clinic Cliniq is a tool that runs health checks before major events, such as upgrades, to determine whether any issues need to be addressed first. You can also run Cliniq routinely to monitor the health of your system. /opt/qradar/support/clinic -h Reports collectGvStats.sh The collectGvStats.sh tool allows you to troubleshoot accumulator issues. Accumulated Data is an aggregate data view used to draw a Time Series graphs or run Scheduled Reports, when you create a search that groups by one or more properties. /opt/qradar/support/collectGvStats.sh -s High Availability (HA) cstate This tool displays the HA cluster status and roles to assist with troubleshooting. /opt/qradar/ha/bin/ha help Troubleshooting defect-inspector The Defect Inspector is a script that leverages a set of fingerprints to detect defects in a log file and display the APAR or defect name. This script helps in quickly checking whether a QRadar system is experiencing an already known issue. /opt/qradar/support/defect-inspector -h Performance findExpensiveCustomRules.sh If it is not tuned properly, custom rules can cause performance issues. This tool allows you to troubleshoot if a rule causes performance issues. /opt/qradar/support/findExpensiveCustomRules.sh -d /root Logs get_logs.sh Collect QRadar logs from a system via the command line interface with the get_logs script. /opt/qradar/support/get_logs.sh -h Logs scrub.pl Do not use. The scrub.pl script is deprecated from QRadar. See log_scrubber.py script for sanitizing logs. Application framework qapp_utils 730.py Do not use. The qapp_utils 730.py script is deprecated from QRadar. See recon for app container management. Network qchange netsetup The qchange netsetup command will assist you in changing the IP address, hostname or DNS server in a Qradar system. qchange netsetup High Availability (HA) qradar nettune.pl This script will assist you in testing the HA crossover connection. /opt/qradar/ha/bin/qradar nettune.pl crossover Application framework recon Recon is a tool designed to aid the troubleshooting of containers and container management on the QRadar Console or App Host. It can allow you to access the command line of your installed applications by using the app container ID. /opt/qradar/support/recon ps Troubleshooting replicationVerify.pl This tool allows to validate if the QRadar configuration database is synchronized across the environment and if it is the same on all the managed hosts. /opt/qradar/support/replicationVerify.pl -h Logs log_scrubber.py To sanitize logs before opening a support case, use the log_scrubber.py utility. This script allows customers to sanitize IP addresses, usernames, hostnames, and domains from logs due to security concerns. The log_scrubber.py script replaces the scrub.pl tool, which is deprecated by support. Updates to log_scrubber.py script are delivered through QRadar automatic updates in the Supportability Tools RPM file. /opt/qradar/support/log_scrubber.py -h Performance threadTop.sh The ThreadTop script can determine which QRadar process is consuming the most resources. This tool monitors QRadar processes and can give an indication of performance issues. /opt/qradar/support/threadTop.sh Administration yum Yum is a software package install manager. Yum can be used in QRadar to manually install RPM files and view detailed version information for installed files, such as DSM, protocols, scanners, and more. yum info DSM-Cisco* yum -y install package filename.rpm Services journalctl journalctl is a logging service similar to a syslog. The command journalctl can be used to display failures or errors from specific services. journalctl -u hostcontext Network tcpdump tcpdump is a common packet analyzer that runs under the command line. It allows the user to display TCP/IP and other packets being transmitted or received over a network. tcpdump -nna0 -i eth0 port 514 -c 4 tcpdump -s 0 -A host 192.168.1.1 and udp port 514 Disk space & partitions df df is a standard Unix command used to display the amount of available disk space for file systems. df -hT Network telnet The telnet command is used for interactive communication with another host using the TELNET protocol. telnet 192.168.1.1 22 Network ifconfig ifconfig is a system administration utility for network interfaces configuration. ifconfig -a Administration rpm Do not use. RPM commands are deprecated from QRadar for installation purposes. See yum for package management and version information. Health systemctl The systemctl is one of the most used commands in QRadar. See the linked tech note for more examples. systemctl start|stop|restart|status Administration wget Do not use wget to download files from Fix Central because files can arrive corrupted if network speeds are not suitable. For direct downloads, use SFTP instead. Administration sftp Use SFTP to download update packages (SFS), installation files (ISO), and auto updates from Fix Central directly to your devices without using an intermediary host. sftp -o StrictHostKeyChecking=no USER_ID@sFTP_SERVER Logs mod_log4j.pl mod_log4j.pl is a CLI and menu driven script that assists users in enabling and disabling debug loggers in /opt/qradar/conf/log4j.xml. /opt/qradar/support/mod_log4j.pl -h High Availability (HA) ha_diagnosis ha_diagnosis is a summary utility that completes a series of tests to output a summary of high availability appliance checks to the administrator. /opt/qradar/ha/bin/ha_diagnosis -h Troubleshooting iteam_support.sh iteam_support.sh is a script that can assist users in general troubleshooting. It can confirm hashes of downloaded DSMs and protocols, troubleshoot performance degradation in the event pipeline, and identify what log source type generated an event based on a QID. /opt/qradar/support/iteam_support.sh Health WinCollectHealthCheck.sh WinCollectHealthCheck.sh runs through a series of tests and automated checks to help validate managed WinCollect deployments. /opt/qradar/support/WinCollectHealthCheck.sh -h Health validate_deployment.sh The validate_deployment.sh script reports when the deployment configuration of the environment is inconsistent, typically meaning the deployment.xml and databases do not have the same entries. /opt/qradar/support/validate_deployment.sh -h QRadar home Return to the QRadar 101 homepage Applications Learn about QRadar apps Deploy changes Learn about deploying changes to QRadar Disk Space Learn about managing QRadar disk space Technotes Browse a directory of our technical notes Software Download software for QRadar Support Assistance Read our support policies WinCollect Learn about WinCollect 7 and 10 Installs and Upgrades Learn about installing and upgrading QRadar Known issues See current and fixed issues with QRadar SSH into your QRadar console as the root user. Run the following command: /opt/qradar/bin/myver -v The Core version field represents the installed version. Interim fix number indicates the interim fix that is installed. If there is no interim fix that is installed, this line is not included in the output. Example output: Product is 'QRadar' Appliance is '3199' Core version is '2021.6.1.20220215133427' Latest version is '2021.6.1.20220215133427' Branded version is " External version is "7.5.0' Branded latest version is " Release name is '7.5.0 UpdatePackage 1' Version installed with is '2021.6.0.20211220195207' Internal version is '2021.6.1.0' RPM version is '2021.6.1.20220215133427' RPM external version is '7.5.0' QRM enabled: 'false' QRM DB enabled: 'false' QVM DB enabled: 'true' QF DB enabled: 'false' Graph DB enabled: 'false' Console: 'true' Console IP: 'x.x.x.x' IP address: 'x.x.x.x' Virtual IP: 'x.x.x.x' Virtual Hostname: 'qr750-3199-22578.cslab.iss.local' Vendor: 'Q1 Labs' Branded Product Name: 'QRadar' Product Description: 'QRadar' Kernel architecture: 'x86_64' CPU supports 64bit: 'true' Operating System: 'Red Hat Enterprise Linux Server release 7.9 (Maipo)' HA identity: 'N/A' Connection to Console is encrypted: 'false' Docker enabled: 'true' Supports Apps: 'true' DNS: 'x.x.x.x', 'x.x.x.x' Internal Hostname: 'xxxxxx.localdeployment' FIPS enabled: 'false' Secure boot status: 'Only available on EFI firmware systems' To check all system in your environment at once This procedure can be used to obtain the information for all systems in your QRadar environment and save it to a file. Note: If you have many devices, the output can be long, and that is why we recommend piping the information to a file. SSH into your QRadar console as the root user. Run the following command: /opt/qradar/support/all_servers.sh -C -k /opt/qradar/bin/myver -v > myver_output.txt Result The output is saved to a file named myver_output.txt in your current directory. [{ "Type": "MASTER", "Line of Business": "{ "code": "LOB24", "label": "Security Software" }, { "code": "BU059", "label": "IBM Software w/o TPS" }, { "code": "SSBQAC", "label": "IBM Security QRadar SIEM" }, { "ARM Category": "{ "code": "a8m0z000000cwsyAAA", "label": "Admin Tasks" }, { "ARM Case Number": " ", "Platform": "{ "code": "PF016", "label": "Linux" } }], "Version": "All Versions" }]