[](http://www.hpe.com)HPE Reference Configuration for Cybersecurity Graph Analytics for Financial Services with HPE Superdome Flex

Combining Graph Analytics with Memory-Driven Computing for detecting cyber threats and anomalies

Complete approval table before submitting paper to Uno for QA. Table to be removed by Uno before publishing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RA ID | Reviewer's Name | BU/Title | Approval required  Y/N | Date Sent | Date Approved |
|  | Steve Tramack |  |  |  |  |
|  | Brett Gibbs |  |  |  |  |
|  | Daniel Pol |  |  |  |  |

Contents

[Executive summary 3](#_Toc19100093)

[Introduction 3](#_Toc19100094)

[Cyber Attack and Cyber Crime 3](#_Toc19100095)

[Data Exfiltration and Cyber Vulnerability Assessment 4](#_Toc19100096)

[Common Cyber Network Incidents 4](#_Toc19100097)

[Cybersecurity Analytics 5](#_Toc19100098)

[Graph Analytics 6](#_Toc19100099)

[Memory-Driven Computing 6](#_Toc19100100)

[Trovares xGT Cybersecurity Graph Analytics Toolkit 7](#_Toc19100101)

[Cybersecurity Graph Analytics – Deployment Architecture 7](#_Toc19100102)

[Scale-Up MDC Architecture 7](#_Toc19100103)

[Cybersecurity Graph Analytics Solution Blueprint 7](#_Toc19100104)

[Cybersecurity Graph Analytics - Proof of Concept 8](#_Toc19100105)

[Prototype Implementation – Deployment Architecture 9](#_Toc19100106)

[Graph Representation of Cyber Network 10](#_Toc19100107)

[Best Practices for implementing scalable Cybersecurity Graph Analytics 11](#_Toc19100108)

[Performance advantages of Scale-Up MDC Architecture 16](#_Toc19100109)

[Cybersecurity Graph Analytics – Solution Configuration and Reference Guide 17](#_Toc19100110)

[Temporal Triangle (TT) Performance Benchmark 17](#_Toc19100111)

[Reference Configuration – Starter , Standard and Enterprise 18](#_Toc19100112)

[Summary 19](#_Toc19100113)

[Resources and additional links 19](#_Toc19100114)

[About Trovares 20](#_Toc19100115)

# Executive summary

IT Organizations must mitigate cybersecurity risk by understanding external access to the infrastructure and how a malware infection or attack in a particular network. Addressing these challenges are massively complex due to complexity of the data center. Between thousands of servers and a web of physical and virtual networks, IT professionals in the financial services sector need a technology that understands this data effortlessly in order to effectively mitigate cybersecurity attacks.

Key business drivers behind Cybersecurity Graph Analytics in Financial Services are

* Bring together isolated Cyber Network data and events to prioritize exposed vulnerabilities
* Derive Known and Unknown Cyber Threats in the context of mission critical infrastructure assets
* Correlate intrusion alerts to known vulnerability paths and suggest best course of action to respond to cyber attack
* Provide cyber threat detection scaling from millions to billions of network devices and associated connections
* Detect Cyber threats in Near Real-Time to reduce high risk of financial data and cost exposure

The HPE Cybersecurity Graph Analytics solution offering for FSI is designed for infinitely scalable cyber network to keep track of massive volume of endpoint vulnerabilities across deployed infrastructure. This solution can help detect largest and richest attack surfaces with frequency of cyber incidents up to 300 times more attacks in FSI than other industries. This solution takes advantage of Superdome Flex Memory-Driven Computing architecture and achieves order of magnitude reduced time taken to detect known threats from hours to seconds.

Goal of this HPE Cybersecurity Graph Analytics Reference Configuration is to identify known cyber threats for defined set of Cyber Threat Patterns including (but not limited to) Lateral Movement Threat Pattern (e.g.: Zombie Reboot Hacking, RDP Hacking, Privilege Escalation Hacking) , Command-and-Control Threat Pattern (e.g.: Connection Proxies Hacking, Remote Service Hacking, Windows Remote Management Hacking) and Data Exfiltration.

**Document purpose:** This reference configuration provides general guidelines for implementing Cybersecurity Graph Analytics workload with HPE Superdome Flex using the Trovares xGT Graph Analytics Toolkit. In addition to outlining the key solution components, this white paper also provides guidelines for configuring and deploying this combined solution.

**Target audience:** The intended audience of this document includes, but is not limited to CSO, CDO’s, LOB’s, Data Engineers, IT managers, pre-sales engineers, services consultants, partner engineers, and customers that are interested in implementation of Cybersecurity Graph Analytics in their existing or new deployments to include Cybersecurity Graph Analytics processing capabilities for their analytic workloads.

This white paper was developed in September 2019.

# Introduction

### Cyber Attack and Cyber Crime

Cyber-attacks have brought a new recognition of the importance of cybersecurity efforts. Attacks have now become widespread, common, and expected in some firms. New attacks are emerging within weeks due to an underground economy that has seen specialists create built-to-sell malware to a waiting list of cyber criminals. The impacts of cyber-attacks have been felt and there are reports that these attacks are only going to get worse. Below are commonly operated crimes across infrastructure network elements.

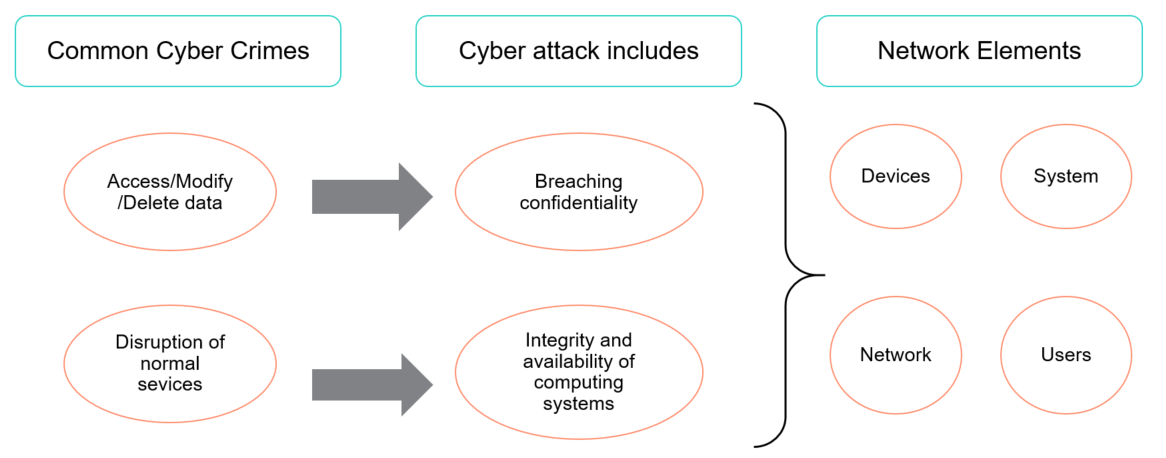


Figure 1. Cyber-attacks and Cyber-crime Concerns

### Data Exfiltration and Cyber Vulnerability Assessment

Data exfiltration is the unauthorized copying, transfer or retrieval of data from a computer or server. Data exfiltration is a malicious activity performed through various different techniques, typically by cybercriminals over the Internet. Data exfiltration is also known as data extrusion, data exportation or data theft.

A vulnerability assessment is the process of defining, identifying, classifying and prioritizing vulnerabilities in computer systems, applications and network infrastructures and providing the organization doing the assessment with the necessary knowledge, awareness and risk background to understand the threats to its environment and react appropriately.

### Common Cyber Network Incidents

A cybersecurity incident or information security incident is defined under ISO/IEC TR 18044:2004 as a **"single or a series of unwanted or unexpected information security events that have a significant probability of compromising business operations and threatening information security”.** When a cybersecurity incident occurs, it very often causes service disruption because one or more IT services might be affected. These IT services might have been destroyed beyond recovery or forced to shut down or the threat actor is monitoring the IT system, preventing employees from continuing their work. In most cases, the threat actors follow a cyberattack kill chain that includes the injection of malware in the network. A cybersecurity incident by definition is either in the past or ongoing.

We can use anomaly-based detection to mitigate DDoS attacks and zero-day outbreaks. DDoS attacks are often used maliciously to consume the resources of hosts and network that would otherwise be used to serve legitimate users. The goal with these types of attacks is to overwhelm the victim network resources, or a system’s resources such as CPU and memory. Below are some very popular cyber incidents that occur frequently.

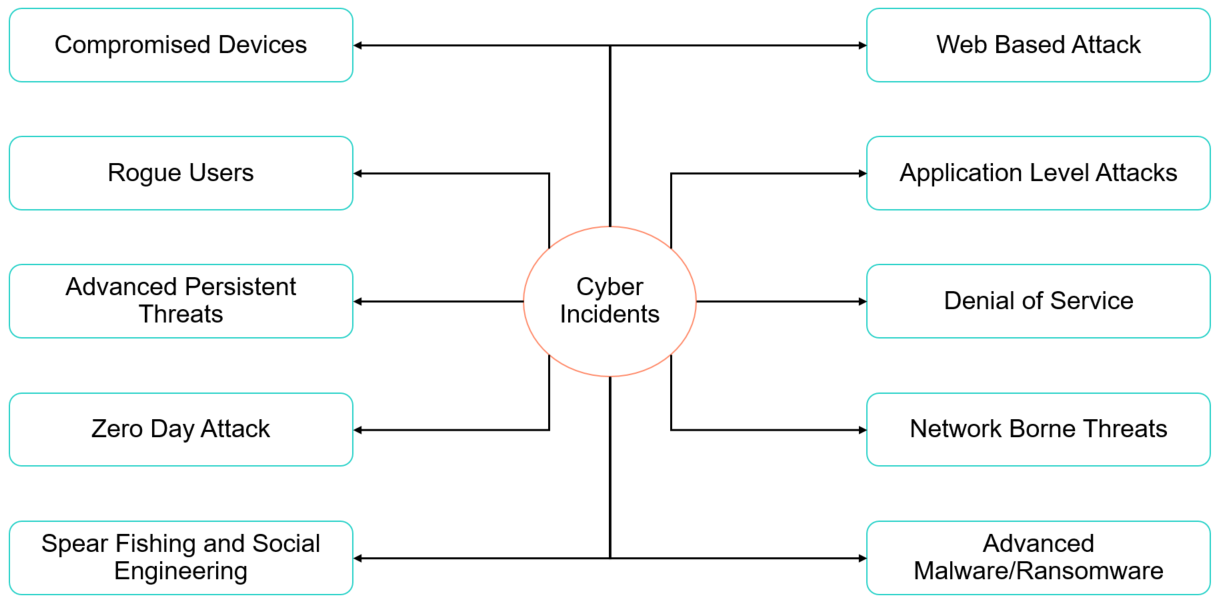


Figure 2. Common Cyber threats in IT Enterprise

1. **Cybercrime** - Any crime that involves the use of a computer as the object of a crime or as an accessory used to commit a crime
2. **Ransomware** - Malware built to extort money from victims by blocking access to their computers and files until they pay a ransom amount
3. **Malware** – Malicious software. There are 3 categories of malwares – virus, worms and Trojans
4. **Social Engineering -** An attack technique that is increasingly being used by cyber criminals to manipulate people into revealing some information or carrying out some actions
5. **Phishing -** A common exploitation attack that involves sending fraudulent emails, that claim to be from reputable sources, to users
6. **Botnet -** A network of zombie devices that have been infected with malware to make them perform certain tasks, such as denial of service attacks
7. **Data breach -** A corporate network is attacked by cyber criminals and some valuable data is stolen.
8. **DDoS attack -** Attackers target a machine with an overwhelming number of requests, thus clogging its bandwidth and ability to respond to legitimate requests
9. **Spyware -** Malware used to spy on people for the purposes of obtaining their personal information, login credentials, or other sensitive information

### Cybersecurity Analytics

Cybersecurity Analytics involve analysis of deployed IT Network to find anomalies. The physical structure of the IT Network shows its topology and the interaction between the network hosts, routers and servers in the form of a graph. A large IP network consists of billions of nodes and numerous edges between these nodes, indicating the traffic between the nodes. This cyber traffic analysis forms the base for Cybersecurity Analytics of the IT Network.

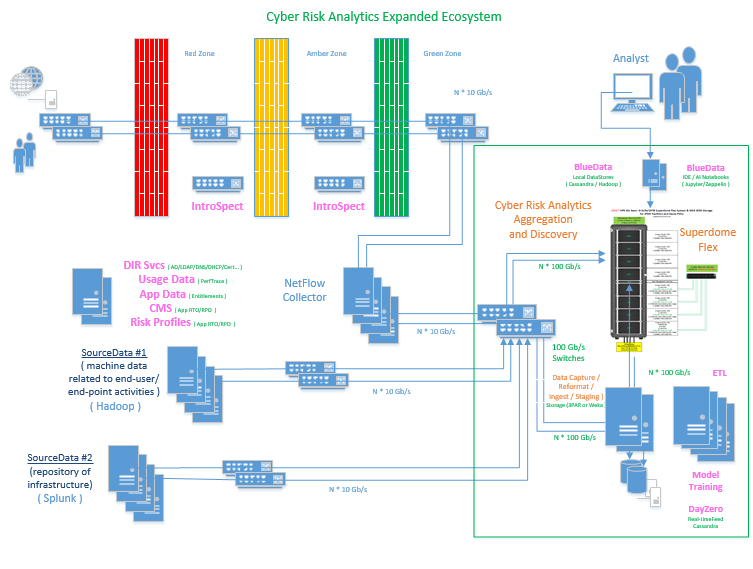


Figure 3. Cybersecurity Analytics deployment landscape

Typical landscape of Cybersecurity Analytics would include following key components

1. Host and Network data from deployed network elements is captured in various logs i.e. netflow log, conn log, dhcp log, http logs, smtp logs etc.
2. Critical data about deployed infrastructure and app would also be available in enterprise end-points like DIR Svcs, Usage Data, App Data, CMS etc
3. Data from these multiple sources would be collected because data aggregation and discovery is required as pre-processing for Cybersecurity Analytics
4. Log data is processed to identify Cyber Network elements and there-by build cyber network graph representing Network Elements
5. Data Scientists and Analytics would interact with In-Memory Cyber Graphs through Multi-tenant Data Science Environment with HPE BlueData ML Ops
6. Cyber threat patterns would be identified and analyzed to detect cyber intrusion behavior
7. Develop pre-trained models in the form of Graph Queries to subsequently leverage for Day-Zero Threat Detection

### Graph Analytics

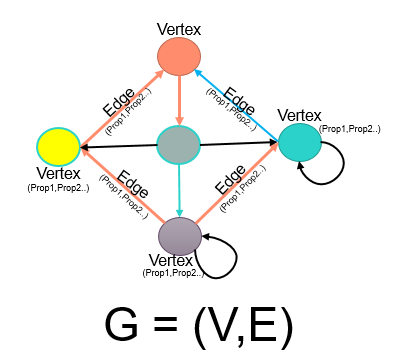


Figure 4. Graph Representation with entities and relationships

Graphs are mathematical structures used to model many types of entities and relationships in physical, biological, social and information systems. A graph consists of nodes or vertices (representing the entities in the system) which are connected by edges (representing relationships between those entities). Graphs, however, are more than just nodes and edges ‐ they are powerful data structures you can use to represent complex dependencies in your data. These Vertices and Edges carry properties describing characteristics of entities and relationships.

Graph Algorithms or Graph Analytics are analytic tools used to determine strength and direction of relationships between objects in a graph. The focus of graph analytics is on pairwise relationship between two objects at a time and structural characteristics of the graph as a whole.

### Memory-Driven Computing

Memory-Driven Computing offers every processor in a system access to giant shared pool of memory where as in traditional systems small amount of memory is tethered to each processor. Memory-Driven Computing is an almost infinitely flexible and scalable architecture that can complete any computing task much faster, using much less energy, than conventional systems. The performance of Memory-Driven Computing is possible because now any combination of computing elements can be composed and can communicate at the fastest possible speed.

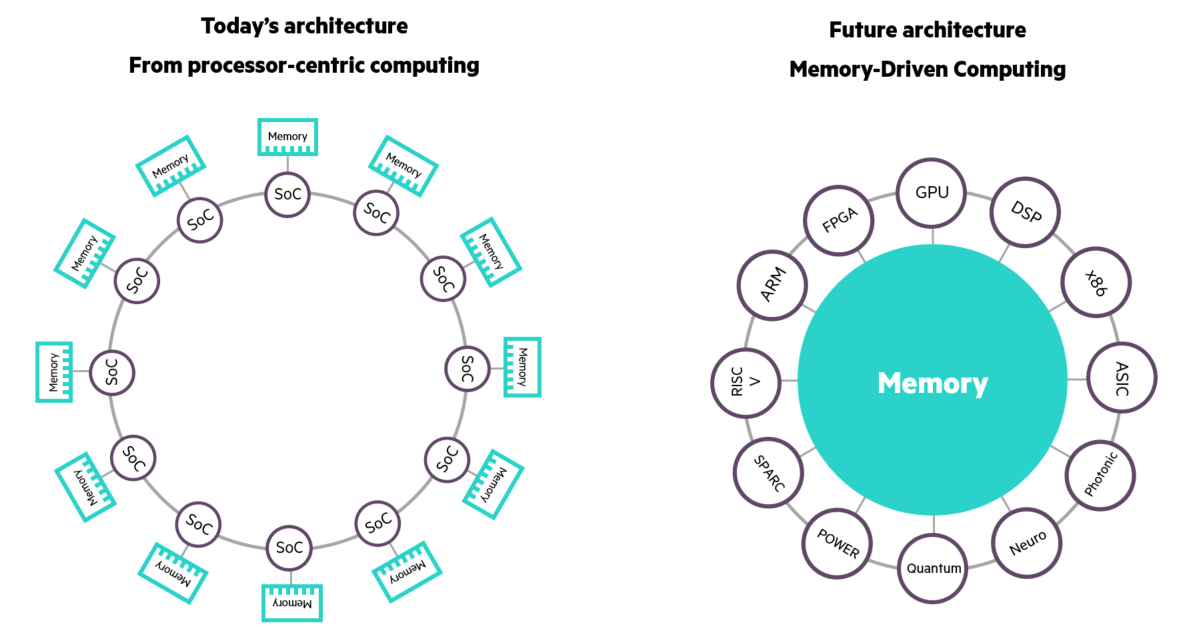


Figure 5. Traditional Processor Driven Computing architecture vs Modern Memory-Driven Computing architecture

# Trovares xGT Cybersecurity Graph Analytics Toolkit

Trovares offers a new type of graph analytics tool which returns search queries hundreds of times faster than conventional graph tools. It supports extremely large in-memory graphs for fast queries. Built on supercomputing technology, Trovares derives maximum performance for cyber, fraud detection, government, and other applications.

It enables the direct ingest of data into the system and avoids database performance issues. Trovares xGT adopts supercomputing techniques such as extreme multithreading and fine-grain locks to achieve orders of magnitude increase in speed and scale.

Trovares xGT offers following key differentiated features for Cybersecurity Graph Analytics

1. Graph representation of Cyber Network Devices
2. Massive parallel processing engine based on Symmetric Multiprocessor (SMP) architecture
3. Optimized search engine for In-Memory processing with Cypher compatible Trovares Query Language (TQL)

# Cybersecurity Graph Analytics – Deployment Architecture

Implementing a scalable Cybersecurity Graph Analytics solution requires understanding the scale of deployment to build cyber network of devices and connection between devices such that insights can be derived to uncover an attack pattern.

Below proposed deployment architecture enables bringing together of isolated data and associated events into an overall picture of decision support and situational awareness. This architecture scales infinitely to detect exposed vulnerabilities, mapped to potential threats in the context of mission critical threats.

### Scale-Up MDC Architecture

Cybersecurity Graph Analytics involves identifying cyber intrusion behaviours in a deployed infrastructure comprising of complex network of servers, routers, gateways, storage etc. Developing such Cybersecurity Graph Analytics involves analysing massive volume of Infrastructure network traffic information from network connection logs, http logs, dhcp logs, smtp logs, netflow information etc. and there by establish a network of infrastructure entities and relationships. This is achieved by building a network graph which enables detection of network anomaly patterns leading to identifying threats like Zombie reboot, RDP hacking etc.

Typical size of these network graphs comprises of billions of graph nodes and properties and relationship between graph nodes. Deriving anomaly patterns across these billions of nodes in real-time requires existence of entire graph in-memory with TBs of large memory infrastructure.

Cybersecurity Graph Analytics use-case with Graph Analytics is ideal for memory-driven analytics.

### Cybersecurity Graph Analytics Solution Blueprint

Conventional graph tools limit companies to sampling a slice of their log data rather than the whole thing, or querying the data over days, not hours. In both cases companies expose themselves to undue risk.

Trovares’ xGT is a property graph engine designed to support extremely large in-memory graphs. Trovares xGT analytics tool provides large graph search capabilities for data scientists and business analysts. With rapid ingest rates and the ability to execute complex queries driven by Python scripts, Trovares xGT provides unprecedented performance and scalability, especially when processing very large data.

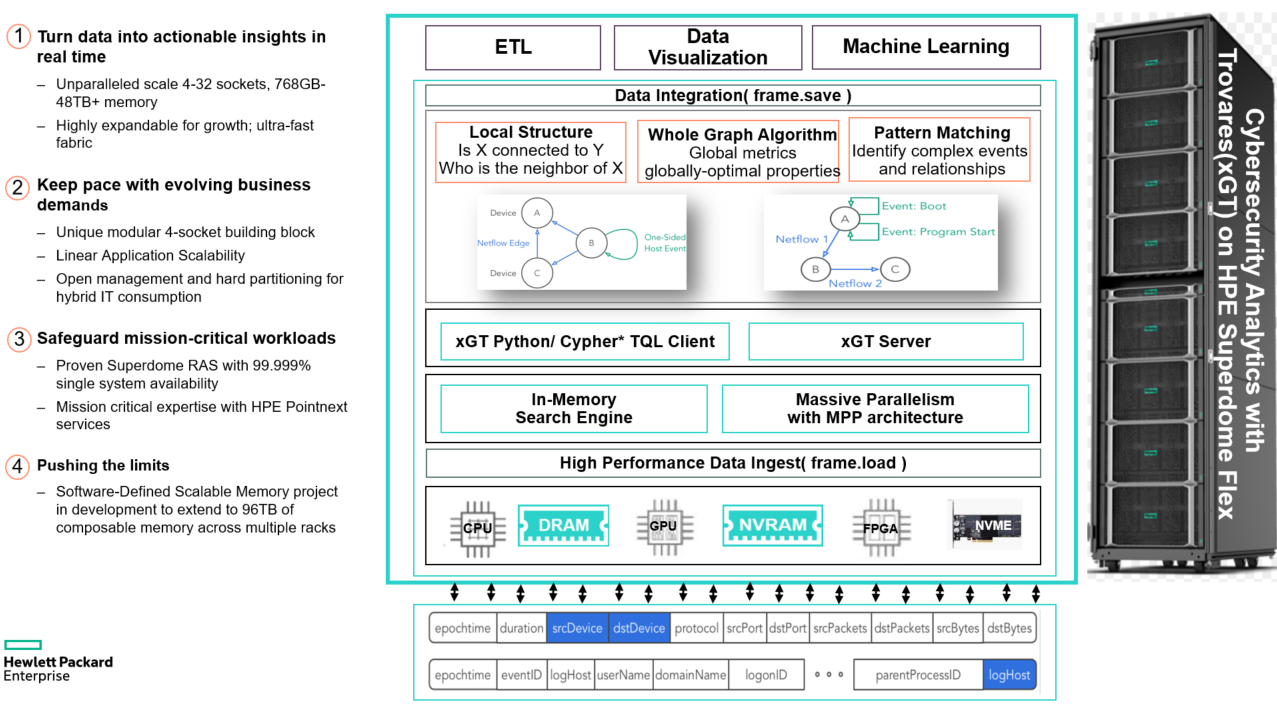


Figure 6. Cybersecurity Graph Analytics with Trovares xGT on HPE Superdome Flex Solution Architecture

Above proposed solution offers following key differentiated features

1. Trovares xGT tool reads massive amount of data into RAM for performing fast *pattern search operation.* There are two ways to get the data into Trovares xGT i.e. across a network or from a filesystem. Fetch such data can be achieved leveraging load() function in Trovares xGT module.
2. Once the data is represented in the form of In-Memory Graph model (Local Structure), each vertex uniquely identifies Cyber Network elements and devices represented by **vertex\_frames**. The connection between these devices represent exchange of network traffic and these relationships represented as Edges also carry properties representing the nature of network traffic represented by **edge\_frames**.

MATCH <structure>

WHERE <optional constraints or properties>

SET <optional property modifications>

MERGE <optional addition of vertices>

CREATE <optional addition of vertices and edges>

DETACH DELETE <optional deletion of vertices>

DELETE <optional deletion of edges>

RETURN <description of answer set>

INTO <results table name>

1. Trovares Query Language (TQL) uses a subset of cypher language to express queries. Trovares xGT offers strongly typed graph elements (fixed schema) with cypher language based Trovares Query Language. Typical query subset is represented as above

# Cybersecurity Graph Analytics - Proof of Concept

Let us look at how above proposed In-Memory Cybersecurity Graph Analytics Solution Blueprint can be deployed to identify cyber threats in operational network of Financial Services Organization.

Use-Case considered for this proof of concept is to detect network anomalies by identifying zombie reboot and RDP hacking events in a deployed cyber network. The attack pattern for such network hack is called Lateral Movement attack.

Lateral movement is a cyberattack pattern that describes how an adversary leverages a single foothold to compromise other systems within a network. Identifying and stopping lateral movement is an important step in controlling the damage from a breach, and also plays a role in forensic analysis of a cyberattack, helping to identify its source and reconstruct what happened.

RDP hijacking is actually a family of attacks, each with different characteristics on how to attain the privileges required to perform the RDP Hijacking. The attack broadly looks like this:

1. Lateral movement starts from a foothold where an adversary already has gained access. We'll call this host A.
2. The attacker uses some privilege escalation technique to attain SYSTEM privilege.
3. The attacker then leverages their SYSTEM privilege to hijack as RDP session to move through a network. The result is to become logged in to another system where the RDP session had been. We'll call this host B.
4. This hijacking action can be repeated to form longer chains of lateral movement; and these chains can be represented as graph patterns:

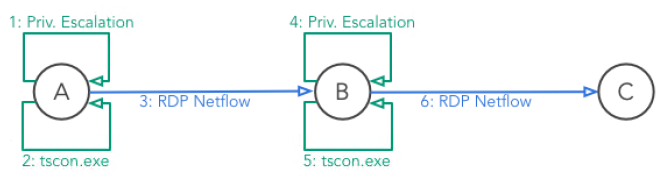


Figure 7. RDP hacking lateral movement attack

### Prototype Implementation – Deployment Architecture

Technology stack considered for this deployment is as below.

|  |  |
| --- | --- |
| **Redhat 7.6** | Linux Distro |
| **Open JDK 1.8** | Java Development Kit |
| **Python 3.0** | Compatible Python Runtime for Interactive Notebook and Graph Analytics |
| **Trovares xGT 1.2** | Graph Analytics Toolkit |
| **SAR Analyzer** | Performance Analyzer for CPU, Memory, Network, I/O Utilization |
| **Jupyter iPython Notebook** | Interactive Notebook for Cybersecurity Graph Analytics |

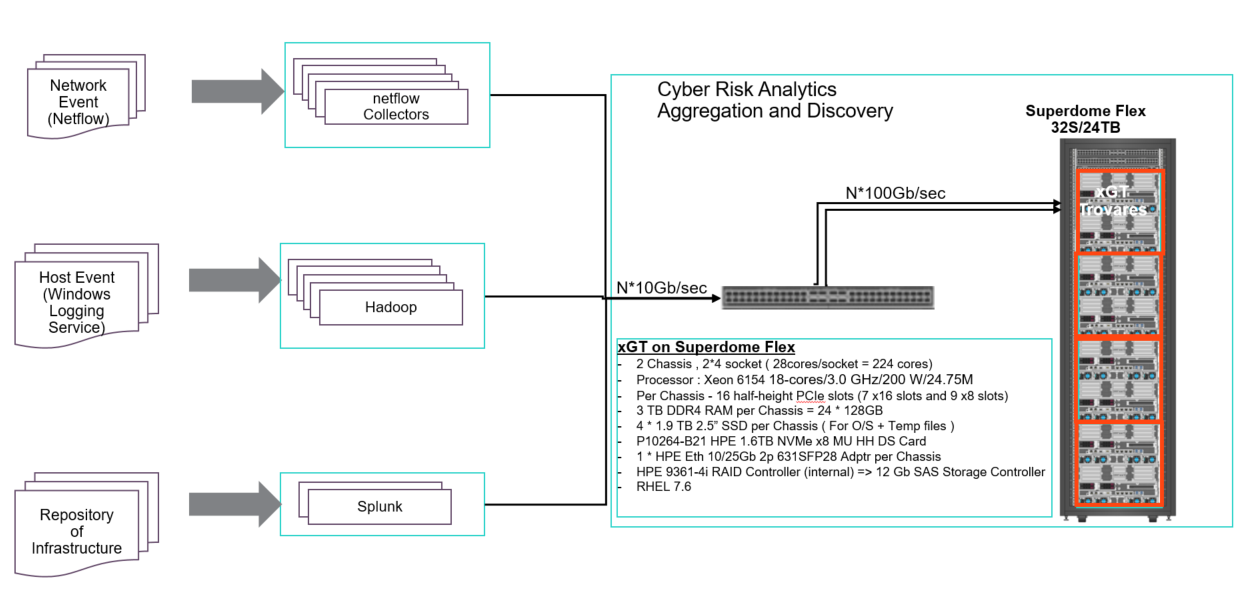


Figure 8. High speed data flow path from different sources into Superdome dome Flex Chassis running Trovares xGT

### Graph Representation of Cyber Network

Steps involved in detection of zombie reboot and RDP hacking events for the identifying Cyber Network attack is as follows

1. 2-Chassis Single Partition environment was created in Superdome Flex system and RHEL 7.6 was installed and repositories were configured to implement Trovares xGT Graph Analytics Toolkit
2. Integrate all the data from multiple data sources and load the data in Trovares xGT Graph Engine. Data loading is performed using load function available in Trovares xGT. Aggregated data is transformed into a Graph Data Model and a network graph is built to represent these network entities with vertex\_frames and edge\_frames.
3. 90 Days of netflow event and host event data loaded and transformed in Graph Data Model in Trovares xGT creates a network graph of 20 Billion Graph Edges (17.9 Billion Netflow Edges and 1.5 Billion log edges) and 212 billion graph edge properties against 3TB of input data from network.
4. Interactive Query operation is performed to detect bot-net behavior over network.
   * Step-1 : Extract the forward RDP Edges
   * Step-2 : Extract Reverse RDP Edges
   * Step-3 : Extract RDPFlow Edge Frames
   * Step-4 : Build Temporal Constraints in RDPFlow Edge Frames

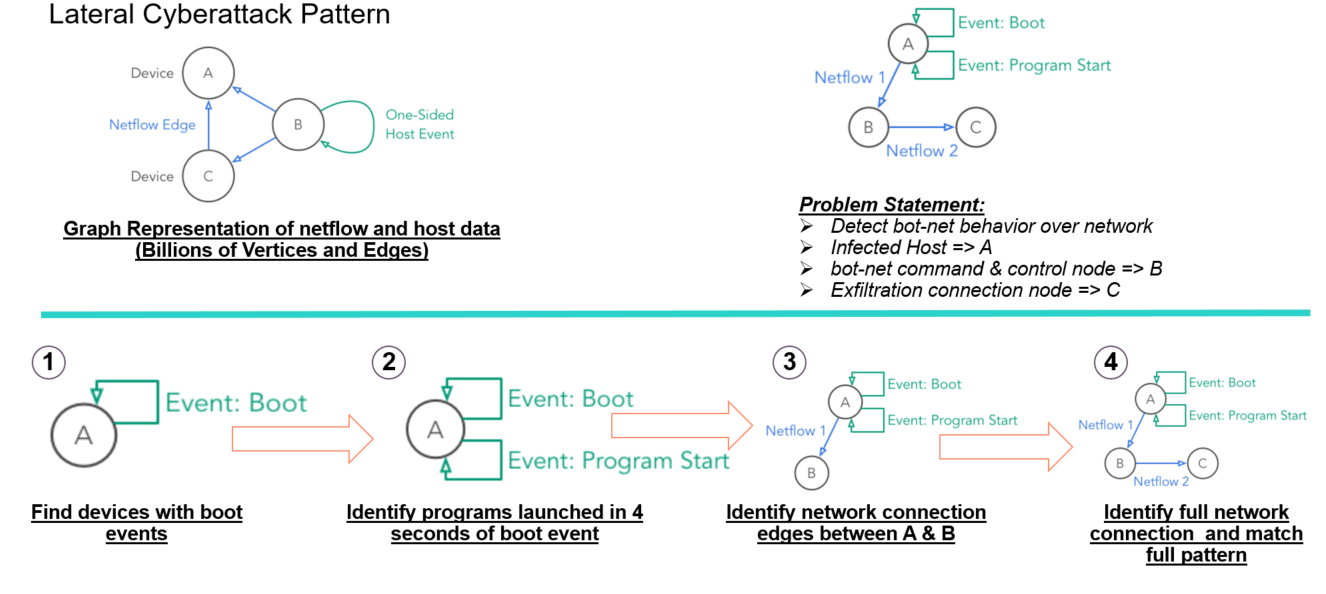


Figure 9. Major steps in graph analytics to detect Lateral Cyberattack pattern

### Best Practices for implementing scalable Cybersecurity Graph Analytics

Following are the best practices adopted while implementing Cybersecurity Graph Analytics with Trovares xGT on HPE Superdome Flex

1. **Validate HPE Superdome Flex partition configuration from Remote Management Console (RMC)**
2. RMC:r001i21c cli> show chassis list

BMCs: 6

BMC Rack UPos Par Power Health CPUs CPU Cores DIMMs Memory (GiB) IO Cards Base-IO

Num Num State Status OK/In OK/In OK/In RAM/PMem OK/In

======== ==== ==== ==== ========== ========== ===== ========= ======= ============ ======== =======

r001i01b 001 01 p0 On OK 4/4 112/112 24/24 3071/0 9/9 Active

r001i06b 001 06 p0 On OK 4/4 112/112 24/24 3072/0 9/9 Present

r001i11b 001 11 p0 On OK 4/4 112/112 24/24 3072/0 9/9 Present

r001i16b 001 16 p0 On OK 4/4 112/112 24/24 3072/0 9/9 Present

r001i23b 001 23 p1 On OK 4/4 112/112 24/24 3071/0 9/9 Active

r001i28b 001 28 p1 On OK 4/4 112/112 24/24 3072/0 10/10 Present

\* OK/In = OK/Installed

1. RMC:r001i21c cli> show npar verbose

This system is nPartition capable

Partitions: 2

Partition 0:

Run State : OS Boot

Health Status : OK

Chassis OK/In : 4/4

CPUs OK/In : 16/16

Cores OK/In : 448/448

DIMMs OK/In : 96/96

IO Cards OK/In : 36/36

Hyper-Threading : On

RAS : On

Boot Chassis : r001i01b

Boot Slots : 3,5

Secure Boot : Off

Secure Boot Next : Off

Volatile Memory : 12287 GiB

Persistent Memory : 0 GiB

Partition 1:

Run State : OS Boot

Health Status : OK

Chassis OK/In : 2/2

CPUs OK/In : 8/8

Cores OK/In : 224/224

DIMMs OK/In : 48/48

IO Cards OK/In : 19/19

Hyper-Threading : On

RAS : On

Boot Chassis : r001i23b

Boot Slots : 3,5

Secure Boot : Off

Secure Boot Next : Off

Volatile Memory : 6143 GiB

Persistent Memory : 0 GiB

\* OK/In = OK/Installed

1. **Install Redhat 7.6 on Partition proposed for Cybersecurity Graph Analytics environment with Trovares xGT on Superdome Flex**

<https://support.hpe.com/hpsc/doc/public/display?docId=a00058577en_us>

1. **Configure yum repositories**
2. Below is an example where RHEL repositories are local:

[RHEL76]

name=RHEL76

baseurl=http://172.30.224.99/RHEL76/

enabled=1

gpgcheck=0

[RHEL76-Supp]

name=RHEL76-Supp

baseurl=http://172.30.1.89:8094/RHEL76-Supp

enabled=1

gpgcheck=0

[EPEL7]

name=EPEL7

baseurl=http://172.30.1.89:8094/EPEL7

enabled=1

gpgcheck=0

1. Below is an example where HPEFS repositories are local:

[HPEFS]

name=HPEFS

baseurl=http://172.30.224.99/HPEFS/

enabled=1

gpgcheck=0

1. Update Repositories

$ yum update

1. **Configure host post RHEL 7.6 installation completion**
2. Set hostname to a unique name

$ hostnamectl set-hostname mdchawkeye.localdomain

1. Verify the hostname

$ hostname -f

1. Edit /etc/hosts with the IP addresses and fully qualified domian name

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4

::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

172.30.224.99 mdchawkeye mdchawkeye.localdomain

1. Edit /etc/sysconfig/network with the FQDN of this host only

HOSTNAME=dl21.n24.epc.hpecic.net

1. **Install HPE Foundation Server (HFS) for monitoring system health and performance**
2. $ yum groupinstall "HPE Foundation Software" ( RHEL)
3. Check memlog service is running once the HPEFS is installed

$ systemctl status memlog

● memlog.service - Corrected machine check interrupt manager daemon

Loaded: loaded (/usr/lib/systemd/system/memlog.service; enabled; vendor preset: enabled)

Active: active (exited) since Tue 2019-08-20 09:01:46 CDT; 2 weeks 1 days ago

Main PID: 56210 (code=exited, status=0/SUCCESS)

Tasks: 0

CGroup: /system.slice/memlog.service

Aug 20 09:01:46 mdchawkeye.localdomain systemd[1]: Starting Corrected machine check interrupt manager daemon...

Aug 20 09:01:46 mdchawkeye.localdomain systemd[1]: Started Corrected machine check interrupt manager daemon.

1. **Mount all nvme disks and enable defaults, nodiratime and noatime in "/etc/fstab" on mdchawkeye**
2. Edit the /etc/fstab file with the below details

#

# /etc/fstab

# Created by anaconda on Thu Jul 25 21:03:17 2019

#

# Accessible filesystems, by reference, are maintained under '/dev/disk'

# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info

#

/dev/mapper/rhel00-root / xfs defaults 0 0

UUID=adc09aaf-8d79-40d6-af21-10d03534bbd2 /boot xfs defaults 0 0

UUID=F538-32ED /boot/efi vfat umask=0077,shortname=winnt 0 0

/dev/mapper/rhel00-home /home xfs defaults 0 0

/dev/mapper/rhel00-swap swap swap defaults 0 0

/dev/nvme0n1 /nvme\_data1 xfs noatime,nodiratime 0 0

/dev/nvme1n1 /nvme\_data2 xfs noatime,nodiratime 0 0

/dev/nvme2n1 /nvme\_data3 xfs noatime,nodiratime 0 0

/dev/nvme3n1 /nvme\_data4 xfs noatime,nodiratime 0 0

/dev/nvme4n1 /nvme\_data5 xfs noatime,nodiratime 0 0

/dev/nvme5n1 /nvme\_data6 xfs noatime,nodiratime 0 0

/dev/nvme6n1 /nvme\_data7 xfs noatime,nodiratime 0 0

/dev/nvme7n1 /nvme\_data8 xfs noatime,nodiratime 0 0

/dev/nvme8n1 /nvme\_data9 xfs noatime,nodiratime 0 0

/dev/nvme9n1 /nvme\_data10 xfs noatime,nodiratime 0 0

/dev/nvme10n1 /nvme\_data11 xfs noatime,nodiratime 0 0

/dev/nvme11n1 /nvme\_data12 xfs noatime,nodiratime 0 0

/dev/nvme12n1 /nvme\_data13 xfs noatime,nodiratime 0 0

/dev/nvme13n1 /nvme\_data14 xfs noatime,nodiratime 0 0

/dev/nvme14n1 /nvme\_data15 xfs noatime,nodiratime 0 0

1. **Set up 8K block size for formatted NVME disks**
2. Execute the following commands

blockdev --setra 8192 /dev/nvme11n1

blockdev --report

blockdev --setra 8192 /dev/nvme0n1

blockdev --setra 8192 /dev/nvme1n1

blockdev --setra 8192 /dev/nvme2n1

blockdev --setra 8192 /dev/nvme3n1

blockdev --setra 8192 /dev/nvme4n1

blockdev --setra 8192 /dev/nvme5n1

blockdev --setra 8192 /dev/nvme6n1

blockdev --setra 8192 /dev/nvme7n1

blockdev --report

blockdev --setra 8192 /dev/nvme8n1

blockdev --setra 8192 /dev/nvme9n1

blockdev --setra 8192 /dev/nvme10n1

blockdev --setra 8192 /dev/nvme11n1

blockdev --setra 8192 /dev/nvme12n1

blockdev --setra 8192 /dev/nvme13n1

blockdev --setra 8192 /dev/nvme14n1

blockdev --report

blockdev --setra 8192 /dev/nvme14n1

1. **Set swappiness to 1 in /etc/sysctl.conf**
2. vm.swappiness=1
3. **Set system kernel parameters**
4. Edit setup\_cpu.sh with following lines

echo -n $"Start Setting kernel parameters on "

echo 250 32000 100 128 > /proc/sys/kernel/sem #This sets SEMMSL, SEMMNS, SEMOPM, SEMMNI

echo 2097152 > /proc/sys/kernel/shmall

echo 2147483648 > /proc/sys/kernel/shmmax

echo 4096 > /proc/sys/kernel/shmmni

echo 65536 > /proc/sys/fs/file-max

echo 1024 65000 > /proc/sys/net/ipv4/ip\_local\_port\_range

echo 4194304 > /proc/sys/net/core/rmem\_default

echo 4194304 > /proc/sys/net/core/rmem\_max

echo 262144 > /proc/sys/net/core/wmem\_default

echo 262144 > /proc/sys/net/core/wmem\_max

1. **Disable firewall**
2. $ systemctl disable firewalld

out: Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.

Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.

1. $ systemctl stop firewalld
2. $ systemctl status firewalld
3. **Setup selinux**
4. Edit /etc/selinux/config file

SELINUX=enforcing ====> SELINUX=permissive

1. **Set a tuned profile network-latency for the server**
2. $ tuned-adm profile network-latency
3. Disable the tuned Service

$ systemctl start tuned

$ tuned-adm off

$ tuned-adm list ( check there are no active profiles)

$ systemctl stop tuned

$ systemctl disable tuned

1. **Disable transparent huge page compaction for good performance improvements**
2. $ echo never > /sys/kernel/mm/transparent\_hugepage/enabled
3. $ echo never > /sys/kernel/mm/transparent\_hugepage/defrag
4. To disable transparent hugepages on reboot, edit and add the below two lines to the /etc/rc.d/rc.local

echo never > /sys/kernel/mm/transparent\_hugepage/enabled

echo never > /sys/kernel/mm/transparent\_hugepage/defrag

1. Modify the permissions of the rc.local file:

$ chmod +x /etc/rc.d/rc.local

1. Add the following line to the GRUB\_CMDLINE\_LINUX options in the /etc/default/grub file:

transparent\_hugepage=never

1. Run the following command:

$ grub2-mkconfig -o /boot/grub2/grub.cfg

1. Set numa balancing in /etc/default/grub file

intel\_idle.max\_cstate = 1

numa\_balancing = disable

1. Validate numa balancing setting

$ numactl -H

1. **Achieve persistency across boot**
2. Edit and Append the following to the kernel boot line in /boot/grub2/grub.conf

elevator=deadline

1. **Disable IPV6 in /etc/sysctl.conf and start rpc bind**
2. Edit /etc/sysctl.conf file with the below lines

net.ipv6.conf.default.disable\_ipv6 = 1

net.ipv6.conf.all.disable\_ipv6 = 1

1. Start rpcbind

$ service rpcbind start

If the above command fails to start, run the following commands

$ dracut -v -f

$ reboot

1. **Set user and file limits**
2. Set User Limits by editing /etc/security/limits.conf file

\* soft nproc 65536

\* hard nproc 65536

\* soft nofile 262144

\* hard nofile 262144

1. Set File Limits by editing /etc/sysctl.conf file

# increase system file descriptor limit

fs.file-max = 65535

#Allow for more PIDs

kernel.pid\_max = 65536

#Disable watch dogs

sysctl kernel.nmi\_watchdog=0

1. **Install JDK**
2. Installing the JDK

$ yum install -y java-1.8.0-openjdk\*

out: Complete!

1. Edit and Add the JAVA HOME and PATH in /etc/profile file to start on system boot

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.181-7.b13.el7.x86\_64

export PATH=$PATH:$JAVA\_HOME/bin

1. Verify the installed version

$ java -version

$ javac

1. **Monitor Main Memory**
2. DIMM Status Check

$ systemctl status memlog

$ memlogd -s

$ grep MEMLOG /var/log/messages

1. Page Migration Check

$ grep "soft offline" /var/log/messages

For Uncorrectable memory errors, /var/log/messages needs to be examined.

1. **CPU Tuning**
2. CPU Performance Tuning

$ cpupower frequency-info -g

$ cpupower frequency-info

$ cpupower frequency-set -g performance

$ cpupower frequency-set -u 3301MHz -d 2000MHz

$ cpupower frequency-info

1. To set parameters automatically during reboot

$ echo "cpupower frequency-set -g performance -u 3301MHz -d 2000MHz" > \

/etc/hpe-auto-config/90\_cpu\_frequency.sh

$ chmod 744 /etc/hpe-auto-config/90\_cpu\_frequency.sh

$ reboot

1. **Linux Kernel Tuning**
2. Parameters set by sysctl

$ /sbin/sysctl -a

Edit /etc/sysctl.conf with the following parameters

kernel.sched\_autogroup\_enabled = 0

kernel.sched\_migration\_cost\_ns = 5000000

kernel.numa\_balancing = 1

vm.min\_free\_kbytes = 512000

scsi\_mod.use\_blk\_mq = 1

dm\_mode.use\_blk\_mq = 1

Cgroup\_disable = cpu

1. **Configure python3 and install Jupyter notebook**
2. Execute the following commands to configure python3

$ yum install gcc openssl-devel bzip2-devel libffi-devel sqlite-devel

$ wget <https://www.python.org/ftp/python/3.7.3/Python-3.7.3.tgz>

$ tar xzf Python-3.7.3.tgz

$ ./configure --enable-optimizations

$ make altinstall

1. Execute the following commands install Jupyter notebook

$ python3.7 -m pip install --upgrade pip

$ python3.7 -m pip install Jupyter

1. **Run Jupyter notebook server component**
2. $ jupyter notebook

### Performance advantages of Scale-Up MDC Architecture

Final query to identify the full network of bot-net behavior in a Cyber Graph of 20 Billion Graph Edges (17.9 Billion Netflow Edges and 1.5 Billion log edges) and 212 billion graph edge properties against 3TB of input data was as below.

MATCH (A)-[rdp1:RDPflow]->(B)-[rdp2:RDPflow]->(C),

(A)-[hijack1:Events1v]->(A)-[privEsc1:Events1v]->(A),

(B)-[hijack2:Events1v]->(B)-[privEsc2:Events1v]->(B)

WHERE A <> B AND B <> C AND A <> C

AND privEsc1.eventID = 4688

AND (privEsc1.processName = "Proc336322.exe" OR privEsc1.processName = "Proc695356.exe")

AND hijack1.eventID = 4688 AND hijack1.processName = "Proc249569.exe"

AND privEsc2.eventID = 4688

AND (privEsc2.processName = "Proc336322.exe" OR privEsc2.processName = "Proc695356.exe")

AND hijack2.eventID = 4688 AND hijack2.processName = "Proc249569.exe"

// Check time constraints on the overall pattern

AND rdp1.epochtime <= rdp2.epochtime

AND rdp2.epochtime - rdp1.epochtime < {0}

// Check time constraints on step from A to B

AND privEsc1.epochtime <= hijack1.epochtime

AND hijack1.epochtime <= rdp1.epochtime

AND rdp1.epochtime - hijack1.epochtime < {1}

AND rdp1.epochtime - privEsc1.epochtime < {2}

// Check time constraints on step from B to C

AND privEsc2.epochtime <= hijack2.epochtime

AND hijack2.epochtime <= rdp2.epochtime

AND rdp2.epochtime - hijack2.epochtime < {1}

AND rdp2.epochtime - privEsc2.epochtime < {2}

RETURN rdp1.srcDevice, rdp1.dstDevice, rdp1.epochtime, rdp2.dstDevice, rdp2.epochtime

""".format(time\_threshold\_between\_step, time\_threshold\_hijack, time\_threshold\_one\_step)

answer\_table = run\_query(q)

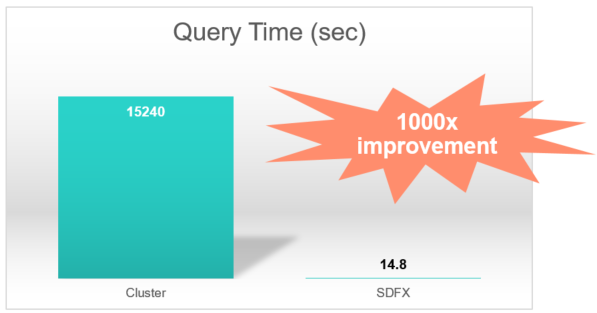


Figure 10. Cybersecurity analytics query performance: Trovares xGT on Superdome Flex vs against scale-out cluster

# Cybersecurity Graph Analytics – Solution Configuration and Reference Guide

Cyber threat anomaly detection requires monitoring and management of cyber network devices across multiple departments. In order to achieve scalable deployment, preferred approach is to understand how Graph Search performance scales.

### Temporal Triangle (TT) Performance Benchmark

This is a synthetic dataset used as a benchmark kernel for assessing graph search performance.

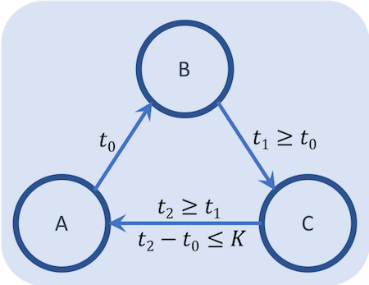


Figure 11. Triangle sub-graph with increasing edge time-stamp property

This dataset contains distribute Graph Generator which has following tools

1. RMAT Graph Generator – Parallel RMAT Graph Generator
2. Graph 500 – Benchmark aimed at assessing Graph Algorithm
3. Attach a single property to each edge i.e. “time-stamp”

Goal of this benchmark is to find all patterns in the generated graph that have a directed 3-cycle where the timestamps on the 3 edges are in monotonically increasing order around the cycle. That is, the timestamps are non-decreasing around the cycle.

MATCH (a)-[e0]->(b)-[e1]->(c)-[e2]->(a)

WHERE a <> b AND b <> c AND a <> c

AND e0.time <= e1.time

AND e1.time <= e2.time

AND e2.time - e0.time < 42

RETURN a, e0.time AS t0, b, e1.time AS t1, c, e2.time AS t2

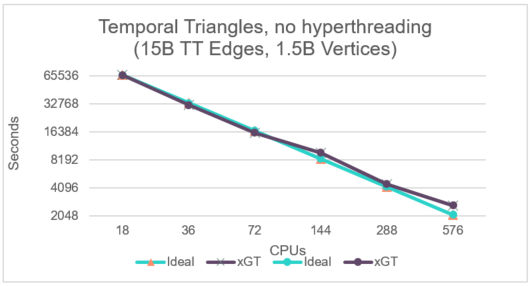


Figure 12. Linear scaling of Temporal Triangle query with scaling processors in Superdome Flex

Graph above highlights linear scaling of Temporal Triangle Benchmark query response time taking advantage of growing no. of compute in multi-socket Superdome Flex environment. Trovares xGT is capable to achieve this because of massive parallel processing architecture based Trovares xGT Search Engine taking advantage of Superdome Flex Symmetric Multiprocessing (SPM) architecture

### Reference Configuration – Starter , Standard and Enterprise

The configuration guide below provides the Graph Analytics S/W framework and infrastructure foundation needed for Cyber Threat anomaly detection to detect wide range of Cyber threat patterns. By leveraging these scalable configuration blueprints customer requirements can be designed for various threat pattern detection. This section builds on the scaling configuration model based on performance scale achieved in Temporal Triangle benchmark highlighted above.

The sizing of the configurations outlined in this section is driven by Size of Cyber Network Graph, Traversed Edges per Second (TEPS) requirements and Threat Pattern Anomaly Detection requirements

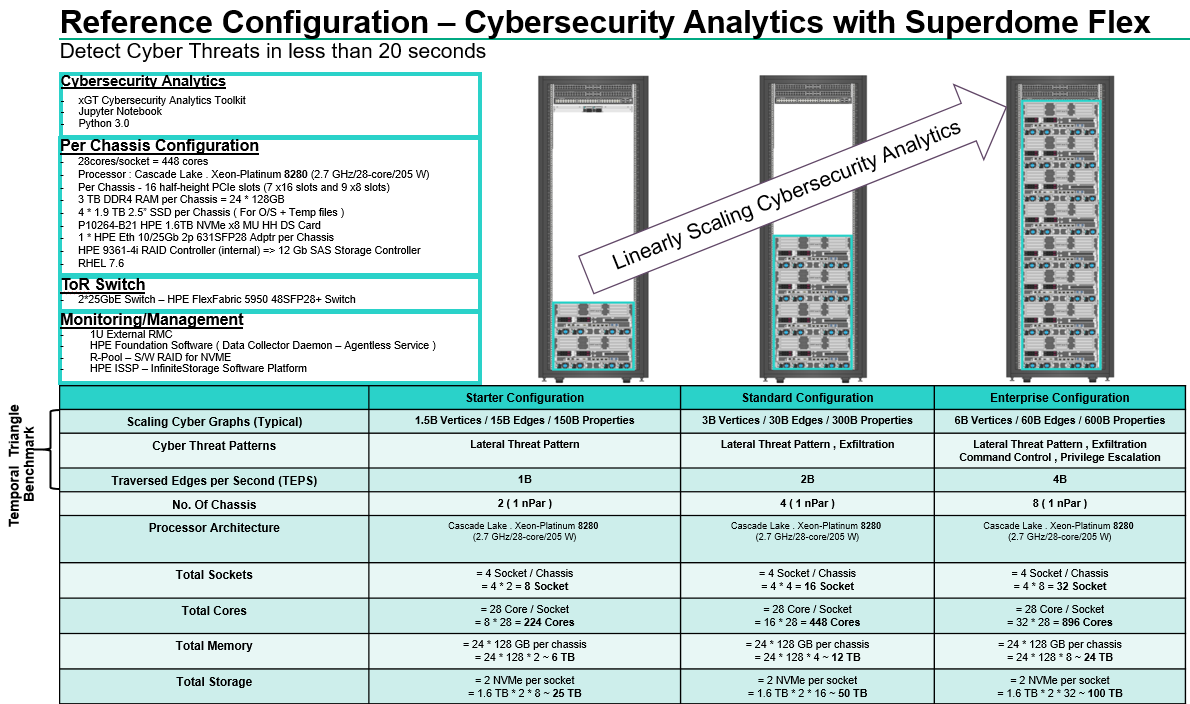


Figure 13. Reference Configuration – Cybersecurity Analytics with Trovares xGT Graph Analytics on Superdome Flex

# Summary

Hewlett Packard Enterprise and Trovares xGT allow one to detect Known Cyber Threat Patterns with massive scale Cybersecurity Graph Analytics Toolkit of Trovares xGT implemented with Symmetric Multiprocessing Memory-Driven Computing architecture of Superdome Flex by building scalable cyber graphs in-memory and finding threat patterns. However, designing and building scalable Cyber Graph can be both complex and time consuming. This white paper provides a Reference Configuration for deploying Trovares xGT on HPE Superdome Flex infrastructure and management software. These configurations leverage HPE servers, storage and networking, along with integrated management software and bundled support. In addition, this white paper has been created to assist in the rapid design and deployment of Cyber Graphs on HPE Superdome Flex infrastructure for graphs of various sizes.

# Resources and additional links

6-cybersecurity Mega Trends, [hpe.com/us/en/insights/articles/6-security-megatrends-1905.html](https://www.hpe.com/us/en/insights/articles/6-security-megatrends-1905.html)

Memory Drive Computing in Superdome Flex, [hpe.com/us/en/newsroom/blog-post/2017/05/memory-driven-computing-explained.html](https://www.hpe.com/us/en/newsroom/blog-post/2017/05/memory-driven-computing-explained.html)

Mission critical Infrastructure for Data Driven Enterprise, [hpe.com/hpe-external-resources/a00037000-7999/enw/a00037029?resourceTitle=Mission-critical+infrastructure+for+the+data-driven+enterprise](https://www.hpe.com/hpe-external-resources/a00037000-7999/enw/a00037029?resourceTitle=Mission-critical+infrastructure+for+the+data-driven+enterprise)

HPE Performance Cluster Manager <http://www.hpe.com/software/hpcm>

Trovares, [trovares.com](https://www.mapr.com/)

Trovares, <http://docs.trovares.com/1.2.0/>

HPE Superdome Flex servers, <https://www.hpe.com/us/en/servers/superdome.html>

HPE Networking, [hpe.com/networking](http://www.hpe.com/networking)

HPE Pointnext, [hpe.com/services](http://www.hpe.com/services)

HPE Education Services, <http://h10076.www1.hpe.com/ww/en/training/portfolio/bigdata.html>

To help us improve our documents, please provide feedback at [hpe.com/contact/feedback](http://www.hpe.com/contact/feedback).

# About Trovares

Trovares is a Seattle-based company creating a new generation of property graph analytics tools. The company’s roots are in the supercomputing space with major players from Cray guiding Trovares into a new era of enterprise problem-solving. For more information, visit [Trovares.com](http://www.mapr.com/)