SOLUTIONS NOTE

Open XDR Security Platform High Availability (HA)

The Importance of Data Availability

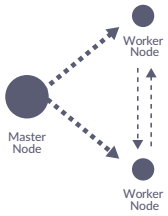
An Open XDR platform plays a significant role in an enterprise's security and risk management, and it's vital to maintain operation and not lose visibility into breach attempts in events such as a power, network or system outage. **At Stellar Cyber, we realize the importance of high availability and have built multiple approaches to ensuring service interruption and data loss are mitigated.** This Solutions Note will give you a basic understanding of how high availability is achieved on our Open XDR Security Platform. You are encouraged to review and implement one or more approaches to meet the HA requirements in your environment.

STELLAR CYBER®

Cloud Native Architecture

It is important to understand the architecture of Stellar Cyber's Open XDR security platform in the context of HA. Our Open XDR platform consists of a family of sensors for collecting data, and a centralized data processor for processing and storing the data. Depending on customer requirements and size of deployment, sensors and the data processor can be deployed on the same physical server, or separately with distributed sensors across the network while the data processor is centralized in a private data center or a public cloud.

The data processor is built upon a cloud native microservice architecture with clustering. It leverages Containers and Kubernetes as the building blocks for such an architecture. This architecture enables auto-healing functionality if any of the micro-services has issues inside the data processor, with real-time health checks and container management.



HA with Node Clustering

The data processor can be deployed as a cluster of nodes to increase performance, provide node redundancy and increase service availability. **A data processor cluster may consist of a single master node and multiple worker nodes.** The master node acts as a load balancer and distributes data to available worker nodes. Once a worker node goes offline, the master node will become aware of this failure and distribute the data to other online worker nodes. A cluster is built to survive loss of worker nodes.

If the master node goes offline, the system would rely on data buffering provided by the network sensors until the master node is brought online again or the standby node becomes active (please see the warm standby section).

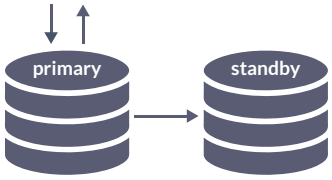
Because the data processor has a clustering technology natively built in as described above, although it can be deployed in a single system, it is strongly suggested to deploy it in multiple systems as a cluster to meet the HA requirements.

DR

Disaster Recovery

Open XDR supports configuration/data backup and restore on another system. **It is extremely important to set this up if you have a single system for the data processor or need an offsite copy of the data.** You can have a data backup and a configuration backup, or a single combined data/configuration backup. If you have separate data and configuration backups, you can configure a different frequency for each. Note that the data backup is a very heavy operation and will have performance impact on the system and be better scheduled at off-peak time.

Although the backup process can be automated at the preconfigured frequency, the restore is a manual process and the time it takes depends on the volume of the data to be restored.



Master Node Warm Standby

In a warm standby configuration, a node can be added to an active cluster, i.e. the primary system, to become a standby system for master node redundancy. After setup, system configurations are automatically backed up from the primary system to the standby system. Upon detection of failure of the master node, the warm standby system can become the primary in minutes after activated manually. This greatly lowers the possibility of platform downtime. There is no additional software license required for this standby system.

Data Replication

In a clustered deployment, the data processor can be configured to keep two copies of the same data across different worker nodes to protect against data loss when one of the nodes is lost. Although data replication is great for data availability, it does have a performance impact on the entire cluster, which accounts for roughly 33 percent reduction in data process capacity and 50 percent in storage capacity when replication is enabled.

Data Buffering

In a distributed deployment, if the sensors that are collecting data lose communication with the data processor due to a network connectivity loss, the sensors can start buffering data to on-board disks and data will be stored locally, based on how much storage has been configured for data buffering. The sensor will continue to send heartbeat checks to the data processor, and once it sees that connectivity has been restored, the sensor will slowly start to transmit its buffer to the data processor so not to overload the data processor with a surge of data after coming back online. **This smart approach ensures data is always available to monitor breaches even if there is a network connection issue.**

Cold Storage

Not all data is equal. For performance and cost reasons, not all the data should be stored as “hot” data which is accessible by the data processor at any time. By default, the hot data is kept for 30 days maximum. You can choose to store older data from your data processor on another server which provides cold storage for a long time period, so that you can re-analyze it later or keep it within reach for compliance reasons. You can import the stored cold data to your working data processor or to a dedicated forensic data processor. This would allow you to visualize and interact with the older data any time with full functionality of the platform at hand. Note that cold storage is different from the data backup and restore for the disaster recovery which is for the hot data.

In-Service Upgrades

Interruptions in service sometimes can be intentional, such as taking a system down to perform a software upgrade to the latest version. If a software upgrade were to take 30 minutes, for example, it could create a scenario where a hacker could gain entry undetected during this upgrade period. **Stellar Cyber’s built-in “In-Service Upgrade” features allow a software upgrade to be performed at the same time that data ingestion is occurring.** This is achieved by the data processor’s micro-services architecture, which allows isolated containers that provide various services to be upgraded separately from data ingestion and data lake containers. The data ingestion and data lake containers are components in the system that are rarely upgraded, and even when they are upgraded, these containers are upgraded last in the process and restart within minutes. This method is much more efficient than upgrading a single component or multiple components all at the same time.