Migrating an Existing Network to OT SDN

Jason Dearien and Tim Watkins
Schweitzer Engineering Laboratories, Inc.

Introduction

Nothing in the networking world is ever static. There are always new technologies, new requirements, or new threats that mean the “old way” things were done is no longer appropriate for any number of reasons. When this happens, it is often necessary to upgrade existing systems to meet these new requirements or make changes to mitigate or address these new requirements.

Many factors affect what must be considered and done based on how extensive the change is. Simply adding a few new devices to a well-designed system is substantially different from taking an existing network and incorporating security controls to protect against and detect cyber attacks. The number of tolerated disruptions in functionality and the allowed downtime during the changeover are big factors that must be carefully considered. Whether the project is a small upgrade or a massive changeover, many of the recommendations in this document should be considered and executed for a successful project.

This paper addresses the challenges of this upgrade process. It explores what must be considered and done along the way to make the upgrade as successful as possible. There are costs and risks involved with making changes to an existing system. This paper is not taking into consideration the physical equipment costs, but instead considers the time and effort spent preparing for, executing, and testing the migration.

Every site has a different tolerance for risk, and upfront costs can be reduced if more risk is accepted during and after the migration. If the installation cannot accept communication problems during and after the migration, then more effort must be spent up front planning the migration. If some disruptions are allowed, a little less time up front can be exchanged for more time troubleshooting problems after the migration.

The goal of this paper is to provide some insight and guidance on what is involved in a network migration so the appropriate balance between risk and cost can be found.

This discussion focuses on upgrading to an operational technology (OT) software-defined networking (SDN) solution. In most cases, the challenges, considerations, decisions, and processes are independent of the final solution.

Shades of Brown

Brownfield is a term used to describe an existing system. This effectively means that "stuff" already exists, and making changes and updates must account for that.

Greenfield, on the other hand, represents a brand-new development or a clean slate.

Upgrading a brownfield Ethernet network is generally believed to be more challenging than building a new greenfield network. Of course, this depends on the extent of the changes, the flexibility to make the changes, and the risk involved with making those changes. These varying factors mean every system is a different shade of brown.

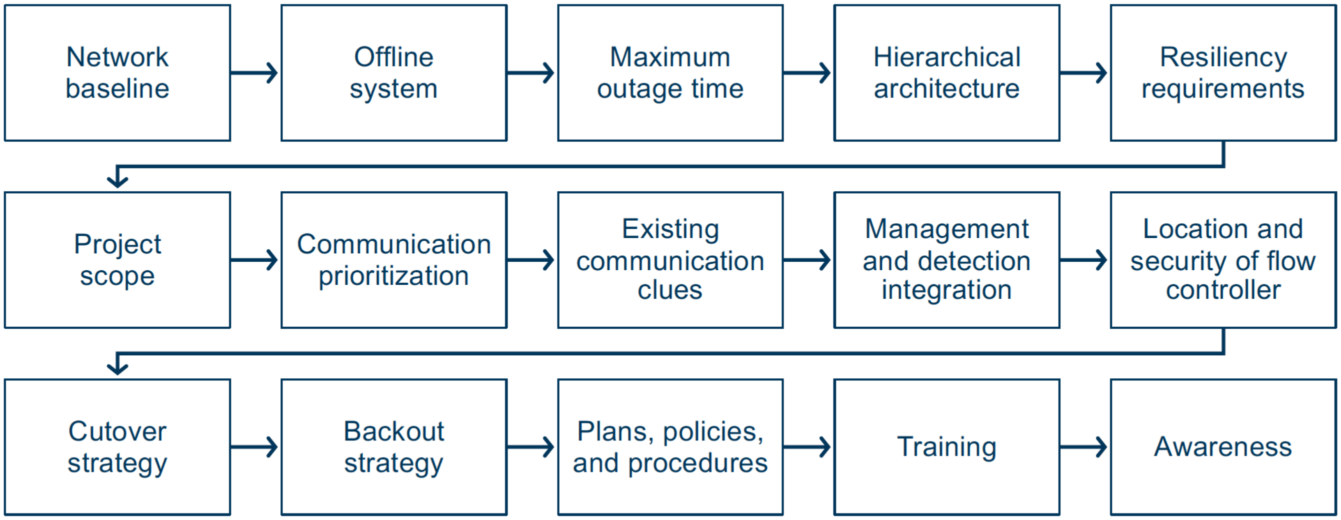
Adding one new Ethernet switch and a few hosts to an existing network would be considered a brownfield change. The network exists, and changes are being made to it. If these hosts follow well-understood policies and behave like existing devices on the network and if the network technology can easily support dropping in a new switch with a few hosts, then this is a minor change and would be a light shade of brown.

On the other hand, if the brownfield network has high-availability requirements (for example, power system devices doing communications-assisted protection), the upgrade is replacing the switches with a different technology, the security posture is changing, and the system must be upgraded with minimal downtime, then this would be a dark shade of brown.

This paper will provide guidance on how to change systems with even the darkest shades of brown. These are complicated upgrades with more at stake for failures. These lessons will still be relevant for lighter shades of brownfield upgrades, but some issues may be of less concern and can therefore be easily addressed or skipped because they do not apply. All the issues should be part of the discussion for a network modification even if they can easily be deemed unimportant.

Path to Success

The rest of this paper will walk through steps that must be taken, or at least considered, in order to successfully navigate an upgrade from a traditional network to a high-performance, cybersecure OT SDN network.


Network baseline
Offline system
Maximum outage time
Hierarchical architecture
Resiliency requirements
Project scope
Communication prioritization
Existing communication clues
Management and detection integration
Location and security of flow controller
Cutover strategy
Backout strategy
Plans, policies, and procedures
Training
Awareness

Preparing for Success

There are many ways to make success much more likely and turn what could be a catastrophic failure into a minor obstacle. Proper preparation and planning can make a huge difference. There are actions that can be taken and information that should be known (or must be gathered if it is unknown) that will increase the likelihood of a successful brownfield network upgrade. It all comes down to how much is known about both the existing network and the new network as well as what failure and success look like.

Many organizations have multiple sites that have similar architectures. The efforts put into the upgrade at one site will pay huge dividends on the upgrades for future installations. The planning, preparation, and the lessons learned can be reused to make future installations go smoother.

INVOLVING ALL THE STAKEHOLDERS

A critical part of the success of the migration is involving the correct stakeholders throughout the planning and deployment processes. Input from a large number of stakeholders not only provides more information to make the best decisions and plans possible, but also engages all the interested parties in the process, gives them a voice, and makes them part of the solution.

Stakeholders for the system may include the following:

- System operators and dispatchers
- Substation, relay, or communication technicians
- Original network engineers
- Business customers
- Safety leadership
- Consumers of the network system data
    - Asset management teams
    - Power system planners
    - Power quality and metering engineers
    - SCADA and automation engineers
    - Protection engineers

EXISTING NETWORK DETAILS

The more that is known about the existing network, the better the migration will go. If all the details are known about the existing network, the migration steps and test plan can be as complete as possible. This scenario will give the best chance for success. On the other hand, if many details about the network are not known, including which devices and services are using the network on a regular or irregular basis, then that information will have to be found or the transition will be much harder. Later in the paper, there is a discussion regarding what information about the network is needed and suggestions about how it can be collected.

A well-designed, secure OT SDN network requires the following system specifics for it to be configured properly:

- Where and how every device is connected (architectural diagram)
    - Which switch(es) and port(s) the device is connected to
    - Whether the device is singly connected, dual-connected in failover, or Parallel Redundancy Protocol (PRP)-connected
    - IP (and possibly media access control [MAC]) address(es) of the interfaces that are used on the network
    - Physical media used for Layer 1 connectivity (fiber or copper)
    - Physical layout and cable routing
- What conversations does each device need to have? (data flow diagram [DFD])
    - DNP3, MMS, or Modbus to the HMI?
    - IEC 61850 GOOSE messaging?
    - Engineering access?
    - Remote Desktop?
    - Etc.

The architectural diagram (physical connectivity) and the DFD (logical communications) are covered in more detail in the Knowing Your Network section.

Traditional Layer 2 networks do not need this level of detailed knowledge because the network will dynamically respond to the locations of the hosts and deliver packets to them dynamically. While this feature makes them convenient, it is also the reason for security vulnerabilities and poor operating characteristics for critical systems.

Knowing all the devices on your network and what conversations they are having is best practice for a well-functioning and secure network, regardless of the technology used to move packets between devices. System operators need the ability to recognize anomalous traffic on the network and perform change management. Without knowing what is allowed, appropriate, and expected on the network, system operators cannot be successful. If all the information is not known, there are two choices:

- Discover and define this information before making the transition to the new network so proper design and test planning can be done up front.
- Discover and define this information during the transition, and react as things are found.

Knowing more up front will make it easier. If you do not know the information, spending the time to discover it before starting the transition is well worth the trouble. This depends on the installation and the people doing the work. It is possible that a successful migration can be accomplished without finding all the details up front and that there is also enough time to react when unknowns come up.

How much you know about the details of the network will directly impact how well the migration goes. Knowing if the migration is truly successful requires even more details and is discussed in the next section.

KNOWING WHAT SUCCESS LOOKS LIKE

On the surface, this sounds easy. If everything is working, then the migration was successful. However, this is overly simplistic. Failing to define success more thoroughly will lead to a longer, drawn-out end stage of your network cutover, and the system could be haunted by small (and potentially large) problems.

If you have a well-defined dashboard that gives you visibility into every set of systems using the network and whether they are currently operating properly, then this dashboard should indicate if everything is working properly. Typically, though, if a dashboard like this exists, it shows only a subset of applications using the network. You may be able to tell that the critical applications used for protection or continuous monitoring are in place and working, but it is unlikely that there is any visibility into the functioning of periodic applications like engineering access or maintenance. Knowing that everything is working properly requires a complete definition of what “everything” is. This can be challenging, and defining it will be discussed in the DFD section later.

There are even more subtle details that must also be considered. The performance and resiliency of the network must be validated. The previous network was operating at a particular capacity and could handle a certain number of failure events and still operate. The following are a few of the design criteria that should be defined and measured to make sure the new solution is satisfactory:

- Packet loss
- Ethernet, IP, TCP, and UDP errors
- End-to-end network latency of applications
- Network stability and uptime
- N – 1 + failure recovery speed
- Network bandwidth utilization
- Critical application priority queuing
- Security standards compliance

After migration, these criteria should be continually measured. They can even be measured more deeply by augmenting the system design with deep packet inspection tools or intrusion detection systems (IDSs). These can show system owners whether the traffic that exists is the traffic that is expected, which is a boon for security monitoring.

WHAT AND WHEN TO TEST

When building and executing the test plan, it is important to remember that the system is not only successful when expected, but that it also fails as expected. Also, make sure to verify that the tests used to validate success are not giving false positives or negatives. Knowing where the failure points in the system are and what a failure looks like is important. For example, if a design expects a certain level of redundancy in the network, test it by inducing failures and making sure that failover paths work as expected.

It is also important to remember, when generating a validation process, to include tests for occasional network uses. Communications that happen all the time, like SCADA polling, are easy to check because if they stop, many systems will notice quickly. The more infrequent network use cases may have no indication of failure until they do not work when needed. Engineering access, backup, recovery, or other human-induced actions are good examples of occasional network uses that need to be tested.

If changes are being made to the network during the upgrade process, consideration should be given to how best to make these changes. One option is to make the changes while making all the other changes and then test everything all at once. Another option is to migrate the existing system over and make sure it is working like it used to and then start making the changes so they can be tested independently. There are legitimate reasons to use either approach. If the new changes will be minor, it might make sense to keep a good baseline with the existing system and then make small changes after the network migration. If the changes will be extensive, it may be a waste of time to migrate the existing system and then make large changes, so doing them all at once may be favored.

STARTING FROM SUCCESS

Before starting system migration, it is important to confirm the existing system is fully operational. It is good practice to generate a test plan and use it against the existing system before any migration happens. If the tests fail, make sure that it is not the test that is in failure. Otherwise, you will be chasing a bad test instead of a broken system, and it will not be clear where the problem is. The best scenario is if the existing system passes all the test cases. Then during migration, the same tests can be exercised to prove success. Of course, this strategy will not work for each new feature or behavior that is added.

Planning the Migration

There are many factors to look at when planning a migration. The system will either be completely unavailable or it will be operating with reduced functionality during the changes. How long your system will be in some degraded state during the migration depends on the system itself, how much preparation was done, and how fast changes can be made. How long the system can stay out of service depends on the criticality of that system, meaning what services it provides that will be missed during the changeover.

If the network is for a power system substation doing communications-assisted protection, then while that network is not operational, some protection functions will continue to operate (because the relays still have hardwired inputs and outputs to make protection decisions). However, without the network, they cannot communicate to make critical system-wide decisions. How long this is acceptable is system-dependent. If the system uses a PRP-based network, one LAN at a time may be able to be migrated without ever losing communications-assisted protection. Of course, operating on a live system and having irregular and/or partial communications may cause certain risks so taking the system completely offline may be more appropriate.

The migration test plan should include checks for success and define what they look like. If everything should be working, even in the middle of the migration, then test that incrementally. If some functionality will be disabled, make sure it is (because if it is operational and should not be, that is concerning), and make sure everything else is working as expected. Finding failures as soon as they happen allows adjustments to quickly be made. If too many steps are taken before testing whether there are failures, the root cause of the failure cannot be easily found.

These decisions must be considered and discussed with the groups responsible for the system to understand what different outage events mean to the day-to-day operations of the system.

SIMILAR BUT DIFFERENT

OT SDN networks move Ethernet packets such that the end devices are unaware there is any difference between them and a traditional Layer 2 switch. The improvement is that OT SDN switches do only what they are told by the controller instead of dynamically making fixed closed-loop decisions like traditional switches. This means even though both OT SDN and traditional Layer 2 switches move Ethernet packets, they decide how to forward packets to the proper destination(s) in different ways and therefore are not directly compatible in a plug-and-play dynamic network.

The ramification is that special considerations must be taken, and often special configurations must be made at the "touch points" where different technologies physically meet and interact. It is not possible to redundantly plug a traditional Layer 2 switch into an OT SDN switch without having the correct configuration on each switch and the correct physical connections. For example, to maintain redundancy in the network between an OT SDN network and a Rapid Spanning Tree Protocol (RSTP) network, the OT SDN network must be configured to move the RSTP Bridge Protocol Data Unit (BPDU) packets in a specific way [1]. Upgrading a network of traditional switches to OT SDN switches requires appropriate configurations in addition to physically swapping them out.

If a network outage can take place with enough time to swap all the switches at one time, the easiest solution is to swap them all out at once. If that is not possible, a special configuration must be constructed for the OT SDN switches to allow them to properly integrate into the traditional switch network while a slower migration happens. Typically, this involves programming them with special flows to allow them to behave more like traditional switches (not completely locked down with more flexible traffic allowed). This more open configuration allows the two different technologies to interact well enough to slowly replace the traditional switches while keeping the network mostly functional. In this state, the network would be operating with some reduced capabilities because neither technology may be able to use its full features due to the mixed-state network. Once all the traditional switches are replaced, the special transitional configuration in the OT SDN switches can be replaced with the more secure and locked-down configuration, which will also include redundant path and failover capabilities once it is a single, homogeneous network.

PRACTICE RUNS

Practice makes perfect. If possible, even on a small scale, practice the migration to prove that the plan works. Work out the kinks, and readjust the plans based on the lessons learned. It is rarely possible to create a complete mockup of the system that is to be migrated. It is a huge advantage to set up a test system that appropriately represents the system applications and services, allowing you to practice by watching how they are impacted and when they are successfully stable and reliable. It should have devices that represent all the critical pieces that have been identified earlier in the planning so how they behave during the migration can be observed. Some section of the existing network—maybe just a few switches—should be constructed, and then the migration plan can be exercised to see if it works. If it does not, or it is cumbersome, make changes to make the process smoother, quicker, and less error-prone.

Doing these tests in a lab environment will make the actual migration go much smoother. The more realistic the practice runs can be, the more valuable they will be. How much time is spent with these mock systems should be directly proportional to the criticality of the system being migrated and/or how much time is allotted for the migration. If the system is critical, getting it right the first time is very important. If there is limited time to make the migration, knowing that the plan will work and how to work through any contingencies quickly is important. If the system is critical and allows for only minimal downtime, a mock system is doubly valuable.

HAVING A ROLLBACK PLAN

Even with the best planning and preparations, things can go wrong. It is highly recommended that a "rollback" plan be created. This rollback plan will be completely different for every installation, but in the simplest form, it involves reverting to a premigration state. If swapping hardware out, this would mean putting the old hardware back in place.

If the end devices did not need any configuration changes to make the changeover, swapping back to the original system might be easy. This is where having a plan in place up front is good. In some cases, the rollback plan might be to leave the old hardware in the rack and powered on until the full cutover is complete and remove it only when everything is successful with the new system. This makes rollback easier.

With staged migrations, a staged rollback plan may be needed. After a certain stage in the migration, rolling all the way back to the beginning may not be necessary. These considerations are system-specific and must be considered during the planning phase.

Having a well-understood go/no-go set of criteria for the migration is critical to success. What these go/no-go criteria are depends on the system and the uptime/availability requirements. Without a plan with clear decision points identified, migration may proceed too far before realizing there is not time to complete the next step or to roll back the current changes to a stable point. Proper planning can help avoid these unfortunate situations.

Knowing Your Network

As mentioned earlier, the more that is known about the network, the better any change will go. Knowing where (and how) all the devices are physically connected and which other devices they communicate with and why makes planning for changes to the new design easier. Without this information, moving to a completely deny-by-default OT SDN network will cause communication problems until this information is correctly identified.

Systems may have some form of network monitoring tools—perhaps an IDS or Security Information and Event Manager (SIEM) where traffic, network state, and events are monitored. These systems can be a treasure trove of information for knowing what is currently on the network. If the system doesn't currently have an IDS or SIEM, designing one into the OT SDN solution is highly recommended. The amount of visibility and information available from the OT SDN network can greatly increase the security posture, visibility, and reliability of the network.

The architectural diagram and the DFD are useful tools to help document all the required information for the network to operate properly. They can also be used to automate the OT SDN network programming. With complete information about the network, generating these two documents (if they do not already exist) should be straightforward. Without complete information, going through the process to generate the documents will help identify what is needed and prompt the proper discussions with stakeholders to gather all the details. The information is required to program the network, so it is helpful to have it documented.

ARCHITECTURAL DIAGRAM

The architectural diagram represents information about the network configuration. First, it covers the physical devices and how they are connected to each other. Every piece of equipment and every Ethernet port that is used for any purpose on the network should be represented in this diagram. The goal is to document exactly what paths are available for the Ethernet network to use to move data between devices. It is important to also know what kind of physical media (copper or fiber ports, single-mode or multimode) is used so that everything will physically plug together. Having different connector types or running out of ports on a switch during an installation can quickly bring the deployment to a halt while new equipment is ordered or other changes are made.

Second, the architectural diagram contains the IP addressing information (and MAC addresses, if desired) for all the devices and ports. After documenting where all the devices are connected, it is necessary to know what IP address will be found on each of the ports used to access that device. Some devices are singly connected, so one IP address may be found on only one port on the network. Some devices have high-availability ports, which means there will be multiple ports on the network that represent that IP address connection. Virtual machines (VMs) are special because they may not be physical devices, but they run on a physical device somewhere. That means that some ports from that physical device are dedicated to that VM and it has IP addresses. How to represent VMs is a personal preference, but to the network they are independently addressable devices, so representing them as physical devices is appropriate.

If devices have multiple physical connections to the network equipment, it is important to understand how those connections are operating in the network. Many devices will have a mode of failover or hot standby for a pair of ports, and this is a very common mode for dual-connected devices. Often referred to as bonding, in this mode both ports are connected but only one is active at any one time. The network is responsible for delivering packets to and from both ports on a failover device so that, regardless of which port is active, communications will work. Another method of high-availability connection for a device is using PRP. In this method, devices are dual-connected but each connection is to a completely independent network, meaning that packets from one network have no path to the other network where the other port is connected. In this case, all traffic is duplicated onto each network and dual-delivered to and from each port. There are subtle differences in configuration for failover versus PRP-connected devices, so it is important to know the method the system will use. PRP requires more specialty devices and considerations.

Third, the architectural diagram includes the reasons why the devices exist on the network. Every device (or VM) in the system is there to serve a purpose. That purpose should have a name, and tagging each device with this name (or names) identifies why it exists. In this paper, we call this name the host profile. In a power control system, there may be breaker relays, motor relays, and perhaps power meters. There is likely an HMI and an engineering workstation. These are examples of host profiles that each device would be tagged with. This abstraction allows talking about devices in more generic terms, and it also means that whether there are a couple power meters or 100 power meters, they will all behave the same and serve the same purpose. It is possible that a single physical device may serve multiple purposes on the network. In that case, the device may be tagged with multiple host profiles. For example, the HMI workstation may also be the engineering access workstation. These are distinctly different functions in the system, but it is possible they would both be performed from the same physical machine. The benefit of these abstractions comes into play when defining the DFD, which is discussed in the next section. This also means functional responsibility between physical devices can easily be moved by simply changing the host profile tags.

The architectural diagram does not have to be a real drawing, but all the information described above needs to be known: what all the devices are, where they are connected, and why they exist on the network. This information could also be represented in a spreadsheet instead of a drawing, as long as it has been written down.

Ideas for collecting this information are discussed in more detail below.

DFD

The OT SDN network is deny-by-default, which means if it is not configured to move certain traffic between specific devices, communication between those devices will not work. The network does not dynamically respond to traffic like traditional networking devices. This is the reason for the significantly improved security posture of the OT SDN network. Also, by clearly defining all required communications up front, it is possible to preplan all the contingency paths in the case of any cable or device failures, which results in significantly improved network performance. These are the reasons the information contained in the DFD is so critical for an OT SDN deployment.

The DFD is where network applications are specified. A communications circuit, in the context of this paper, is all the configuration details and settings required to allow two devices to communicate the information they need across the network. Creating this communications path is called circuit provisioning. The goal of the DFD is to very clearly identify every communications circuit that is required on the network and all the properties of that circuit. For example, if the system has 20 feeder relays and 30 meters and the HMI and SCADA both need to collect data from these devices, that needs to be clearly specified. If the meters use Modbus and the relays use DNP3, that must be specified. If there is an engineering workstation that needs access to some devices with one protocol but uses a different protocol for other devices, that also needs to be detailed.

There are a few constructs of the DFD that are key to understand. The fundamental building blocks are the protocols and the applications.

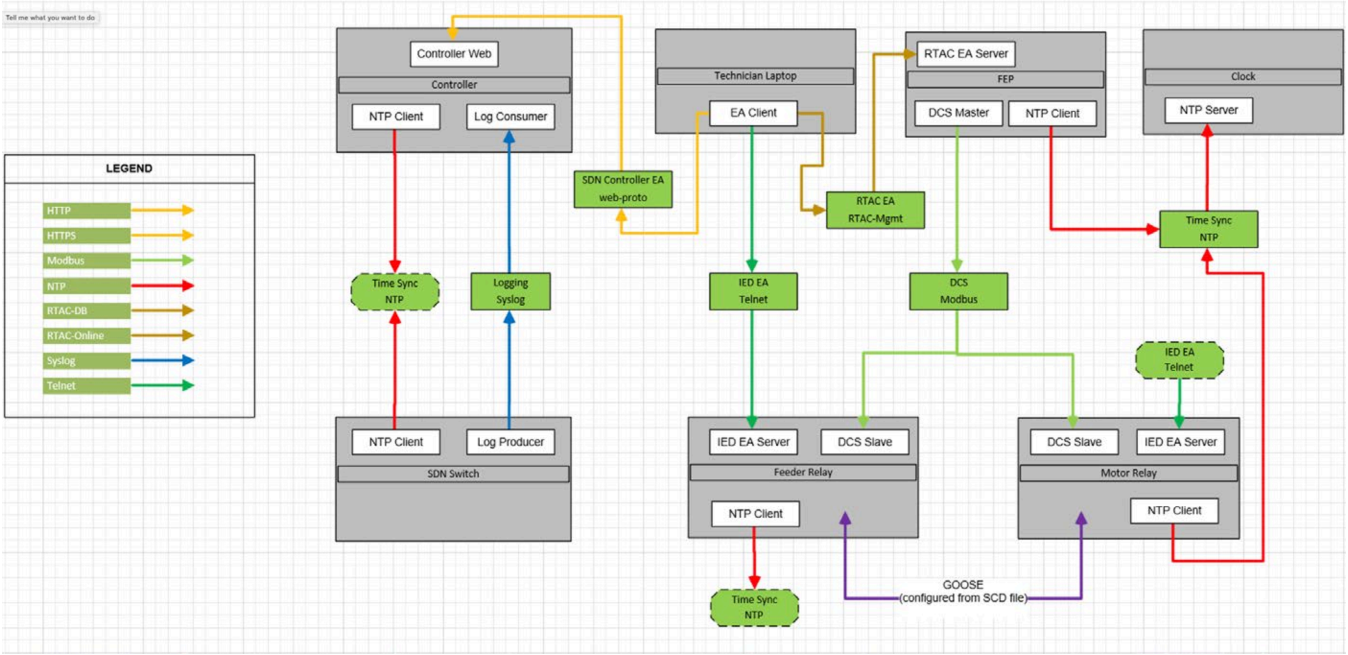
Protocols define the match criteria of the Ethernet packets that will be moved around the network and represent the different conversations between devices. All the details of the protocol must be specified so the OT SDN network can be configured properly to match these packets and move them appropriately. Understanding the details of the network traffic is important when defining protocols, so some research (reading or analysis) may be required to get these protocols defined properly. The more details that are included in the protocol, the more exact the matches will be for the traffic on the network, which will result in tighter security. It is possible to define the protocols loosely to match a larger number of communications between devices, at the sacrifice of security and visibility. Doing so is generally not advised. Details for a protocol will include things like unicast versus multicast, unidirectional versus bidirectional, IPv4 versus GOOSE, and TCP/UDP port numbers. Further details of protocol match criteria are beyond the scope of this document.

Applications are the constructs that tie everything together. An application pulls together a collection of hosts (both source and destinations), a protocol, and a priority. Applications can call out specific hosts for source and destination devices, or this can be abstracted by using host profiles. Applications are given a name and a priority for the traffic on the network and comment what purpose this application satisfies in the system.

With the example above regarding the HMI and SCADA that need to collect data from a system with 20 feeder relays and 30 meters, the following applications might be defined to satisfy those requirements:

Name	Source(s)	Destination(s)	Protocol	Priority
HMI data	HMI	Relays	DNP3	High
HMI data	HMI	Meters	Modbus	High
SCADA data	SCADA clients	Relays	DNP3	Critical
SCADA data	SCADA clients	Meters	Modbus	Critical
Engineering access	Engineering access workstation	Relays, meters	Telnet	Low

Of course, a DFD is meant to be a diagram that represents the information in the table above, instead of a spreadsheet. A diagram representing the data flow can be easier to read, visualize, and review for other people. The following is an example DFD.



NETWORK BANDWIDTH ANALYSIS

Network bandwidth is another topic to consider while defining all the communications that will be allowed on the network. After going through the process of defining the DFD so every possible communications circuit is well known, it is possible to do calculations on how much bandwidth each of those communications circuits needs.

Network utilization can be measured in a couple ways: packets per second (pps) or bits per second (bps). Typically, bps is the most common and is usually measured in megabits per second (Mbps) because most links are rated in Mbps (or gigabits). If a link between two devices is rated at 100 Mbps, a maximum of 100 megabits can be transferred across the links every second.

Each communications circuit between two devices has minimum and maximum bandwidth rates. Some devices that use older, polled SCADA protocols communicate very slowly with reasonably small packet sizes. A 100 Mbps link can handle hundreds of these conversations easily without being burdened. Some protocols (e.g., IEC 61850-9-2 Sampled Values [SV] or HD camera streams) may stream live data/values and can easily use many Mbps per device.

Link saturation occurs when more data needs to be moved across a single network link (a cable between two devices) than the link is designed or configured to move. Anything above that capacity will be discarded.

The purpose of bandwidth analysis is to understand the bandwidth requirements of the devices and protocols on the network and ensure that the necessary bandwidth is available on all the links in the communications path. It is also important to consider the failure modes of the system. Some protocols send more data during an event than others do, and sometimes the network is in a degraded state because of equipment or link failures, so some additional bandwidth may be required on alternate paths that may be shared with other circuits.

Bandwidth analysis is an important part of building a reliable and robust system. If the network is working near the limits of what it can transport when operating normally, it may not handle the required traffic during a failure event. A system that occasionally goes over the limit and loses traffic because of saturation can be difficult to diagnose and produces unreliable behaviors. Having good bandwidth analysis up front can help identify possible trouble areas or give confidence that the system will be robust in all communication situations.

COLLECTING DATA

Having a complete set of diagrams for the system provides confidence in how the network is physically built and what communications it is moving and why. This knowledge will make any modifications or upgrades much easier. The challenge is that if all this information is not already known, it must be discovered.

PHYSICAL LAYOUT

Getting a complete picture of the physical layout of devices can be done in a few ways. One way is to physically visit every device, trace all the cables, and document their connections. The original wiring diagram might be mostly correct and just needs updated. It is convenient to have two or more people do this so one person can trace the physical layout and report it to someone who is doing the documentation.

If the system has been mapped by a network management tool, it may be possible to extract physical architecture information from the tool. It may take the form of the Simple Network Management Protocol (SNMP) IF-MIB walk to collect neighbor data, or perhaps there is a feature-rich graphical interface that shows all the connections. Many networking devices use the Link Layer Discovery Protocol (LLDP) to identify and extract connected network device information. This may be a great place to start, but often there are things that are not depicted, so a physical audit at some level is recommended.

All the physical connections do not provide the configuration information of the devices, so each device's IP, netmask, and gateway (and MAC address, if wanted) must be retrieved and recorded. Hopefully, most of this information is already known and documented somehow, but if not, it must all be collected so it can be used in the configuration.

The following list provides examples of locations to find some of this information:

1. Existing network diagrams
2. MAC tables in existing switches
3. IDSs or SIEMs
4. Packet capture (PCAP)
5. Asset management solutions
6. Preventative maintenance logs
7. Asset procurement logs
8. Other product or vendor features (Cisco Discover Protocol, LLDP, etc.)

Once all this information is collected, it can be combined into a complete architectural diagram of the existing system. Having one for the existing system may not be necessary if the new system will be a complete overhaul and the old system will never be needed. Often, though, the existing system must continue to operate while the new system is being installed. Also, during the migration, if something happens that requires the system to revert to a previous state, it will be very valuable to know what that previous state was.

DFD CREATION FROM CONFIGURATION INFORMATION

Collecting the details of the physical system is often easier than figuring out the details of the communications flow for the DFD. As with the physical system, there are many ways to approach collecting this information, and it is typically a mixture of all the methods that provides the most complete configuration information.

The first method is to look at the configuration of the devices on the system. For example, things like HMIs and SCADA data concentrators must be configured to reach out to the devices they want data from, so that is an easy way to find the source and destinations and identify the details of the protocols to each one. You also need to identify primary and backup servers on the network and make sure their configurations are correct for their roles in the system. Specifically, if one device is backing up another, do they have the same configuration or has one been modified so it no longer matches the other? This configuration collection effort is a great opportunity to do an audit on different systems.

When looking at device configuration to identify what the network is used for, it is important to look at many sections of each device. Of course, this is dependent on the device, but if it is a Microsoft Windows machine, there may be network drives mounted and Microsoft Active Directory connections in addition to the more obvious function of the device in the system. Also, some services, like Remote Desktop, may be enabled, but it may not be clear who is allowed to use the service, so an investigation must be done to determine who uses it. There may be logs on the machine about who connected recently, but this would not guarantee a full list. If the system is well documented and controlled, perhaps this information is in the plan, policies, and procedures for your organization.

During this process, it is likely there will be similarities between devices, especially the IEDs, in how they are configured. At this point, the host profile concept could be applied to a group of devices. Rather than inspecting the configuration for each of them, choose a small sampling instead.

Another challenge with discovering this information from the existing devices is that some devices may be transient. Often, there is an authorized group of people that is allowed to connect laptops into the system to perform certain operations. If the details about this transient access are not well known, gathering this information may be challenging. Conducting interviews with system operators or technicians about what they do on the network and how they do it may provide useful insight.

Of course, any way to automate this audit would eventually save time. Building tools to do a good audit, if they do not already exist, takes longer than doing an audit one time. But once the tools are built, they could be used repeatedly and could quickly pay off that extra expense. The process of automating the collection and categorization of the configuration data will also give insight into what all the devices are really doing. This effort is dependent on the size and complexity of the individual site. Some tools may already exist that can parse the configuration of devices to extract their required communications circuits. Some products provide these configurations in easy-to-access forms or in industry standards like IEC 61850 Substation Configuration Description (SCD) files. Using these sources of information is the best way to get to an accurate description of the communications circuits the devices require the network to provide, because it is specifically called out in their configuration.

DFD CREATION FROM TRAFFIC ANALYSIS

Another way to collect the required details to make a more complete DFD is by analyzing traffic between devices that are actively using the network. There are many methods to get access to these packet streams, and usually you can use a tool like Wireshark to analyze them. There are also challenges and pitfalls with this method. The most notable ones are:

- Only the traffic that is actively on the network at the time is seen by the capture. This means that any transient devices or irregular operations will not be identified. One irregular operation may be engineering access, and missing this circuit would not be catastrophic. However, if a special communications circuit were used to trip a breaker and that was not noticed, the new system would fail exactly when it was needed the most.
- Getting access to the network and altering the configurations to collect the packet capture. Modifying the active switch settings or even getting access into the physical buildings may not always be easy on a live system.

Once traffic has been captured, it must be analyzed and the communications circuits that are needed must be identified. It is very likely that there is traffic that should not be allowed that was previously unknown on the network. Not all the traffic that is identified should always be allowed.

PLANNING FOR FUTURE CHANGES

The success of any migration should not only be measured by the ability to meet the new system demands and requirements but also by the ability to adapt and accept future changes. Following the guidelines found above and completely documenting the system allows this work to be avoided the next time changes are needed. Future modifications can be done more quickly with more confidence as long as the designs are maintained over time.

Conclusion

There are many reasons to upgrade a system to different technology. Perhaps there are new requirements or threats that must be addressed, or an opportunity for new functionality that improves the overall business. Changes in technology and requirements and regulations require periodic system upgrades. Migrating an existing system to new or different technology for any reason has risks. It is important to understand these risks and address them head-on. Completely understanding the existing system and knowing what will be different with the new system helps create a successful migration plan. Spending the time up front to thoroughly identify plans and procedures for testing and deployment are critical.

How much is known about the details of the system will directly impact how well the migration goes. One takeaway from this discussion is that it is critical to know what success looks like. It must be clear what a fully functional network looks like and that it can be confirmed. You must be able to test and verify the operation of all aspects of the system, so you must know what all the applications are and their unique requirements. Generating a complete test plan is a critical step that should not be overlooked.

The completeness of the definition of success and the quality of the test plan will heavily influence the success of the network migration. It is not sufficient to test only the ideal scenario where everything is in a good state. Testing must also include failure cases. This ensures the limits of system stability are identified and understood.

By using the ideas presented in this paper and involving all the stakeholders of the system in the discussions, it is possible to identify all the risks, challenges, and existing or missing data as well as create the plans and procedures to perform a successful migration.

Reference

[1] J. Dearien, "Setting Up a Fully Redundant RSTP-to-SDN Tie Point," SEL Application Guide (AG2017-28), 2017. Available: selinc.com.

Biography

Jason Dearien received his BS from the University of Idaho in 1993. After graduation, he was a founding member of a small startup software contracting business. Later, he was involved in ASIC development at a fabless semiconductor company, working on compression and error correction technologies. In his 20 years at Schweitzer Engineering Laboratories, Inc. (SEL), he has led various product development projects and is presently a principal engineer in the R&D communications department, focusing on network communications with SDN.

SEL SCHWEITZER ENGINEERING LABORATORIES

info@selinc.com | selinc.com | LWP0037-01 | 20220110