

## Interview: Smart Grid Security Expert Justin Searle

With the changing landscape of warfare away from nation-states only utilizing conventional means to the addition of mobile rogue outfits utilizing cyber-attacks, not only countries but also organizations of all shapes and sizes now need to concern themselves with a new threat. Slowly but surely, the real vulnerability to the power grid is starting to grab the attention of both the public and private sectors. Along with that comes more media attention and in turn pressure to make sure these systems don't come crashing down affecting hundreds of millions citizens dependent on today's modern conveniences.

With the need to secure such systems also comes the need for expertise and education. Enter Justin Searle, Managing Partner at UtiliSec. UtiliSec provides security consulting services to utilities and vendors in the energy sector. Some of the services offered include security assessments, guidance on regulatory issues like the NERC CIPs, participation in standards work and security training services. So who better to interview in order to shine a light on some of the many aspects of this burgeoning field of security? Here's several questions to get us all up to speed.

Discuss in Forums {mos\_smf\_discuss:Editor-In-Chief}

1. How were you able to advance in your career far enough to specialize in Smart Grid and SCADA Security?

You know, the funny part is one of my first major jobs was in control systems. Back in High School I worked for almost two years with an engineering firm that designed and built control systems for water treatment facilities. My job was to assemble, wire, and test the control cabinets that housed the switches, indicators, and PLCs.

My original plan was to go into electrical engineering, and I jumped at the opportunity to participate in a work release program that gave me a jump start in the field. However when I started college, I found my attentions pulled in a few more directions. I ended up graduating with a degree in technology education with an emphasis in electrical engineering and computer science. I went on to get a masters degree in International Business and Information Systems.

Career-wise I went in yet another direction. I found myself pulled towards Information Security. As many security professionals, I started on the defensive side. I quickly learned that while I loved the complexity and challenge of the defensive side of IT, it carried with it a moderate yet never-ending stress of trying to get things properly secured. Figuring out how to defend against an attacker was fun, but the continual battle of getting security implemented was not a stress I wanted to deal with in my career. After a couple of years I found myself performing more and more penetration tests, until I ended up doing it full time.

As for my experience with the Smart Grid and SCADA, it started with InGuardians. While working as a Senior Security Analyst on their great team, we had an electrical utility approach us about bringing out penetration testing expertise to the energy sector. We jumped at the opportunity. Since that time, I've performed numerous security assessments and penetration tests for electric utilities and the vendors that sell equipment in that market. I've also played key roles in the creation of several industry accepted documents like the NIST Interagency Report 7628, several security profiles in the Advanced Security Acceleration Project for the Smart Grid (ASAP-SG), and am still heavily involved with National Electric Sector Cybersecurity Organization Resources (NESCOR), and Smart Grid Interoperability Panel (SGIP).

2. SCADA is an acronym for supervisory control and data acquisition, but it is commonly used to refer to all control systems used in industrial facilities like those that maintain the power grid. Could you go over specific terms such as DCS, ICS, HMI, PLC, RTU et al including SCADA to make sure that the readers are using the correct terminology?

Sure. However let me preface this explanation with the warning that I'm going to attempt to draw a clear picture for this site's primary audience, that of IT Security professionals. The parallels I draw and examples I give may not be the ones that control system engineers would agree with, however my explanation should facilitate bridging and IT person's existing knowledge with that of a control system.

Let's start with the broadest term and work our way down. Industrial Control Systems or ICS is the broadest of the bunch encompassing most of the industry that run systems (not necessarily computers) that control automated and semi-automated industrial processes. ICS can be broken down into several sub-categories such as energy (electricity, oil, gas, nuclear, etc&hellip;), chemical, water (treatment and waste), manufacturing, and many others.

These industry sectors use technologies like distributed control systems (DCS) and supervisory control and data acquisition (SCADA) to monitor and remotely control their industrial processes. While traditionally there was a distinct difference between DCS and SCADA, they differ somewhat between each industrial sector and in many cases have faded over time. For the purposes of this article's audience, it is usually safe to assume they are the same.

If you were to step into an ICS Control Center, the most visible piece of SCADA equipment would be the human machine interface (HMI). The control operator monitors this data and initiates control commands to this user interface. HMIs can be single purpose machines, traditional applications installed on workstations running Windows (and sometimes other commodity OSes), and we are starting to see HMIs being built as web applications. These HMIs speak to the SCADA controlling server that is usually installed back in the ICS data center. This server is also sometimes called the acquisition server. It collects the data from and sends control signals to the process controlling devices. Another system we have communicating with this SCADA server is the historian or a series of historians. The historian is basically a database that the SCADA server pushes data to and in some cases pulls data from. One of the primary architectural reasons for the historian is to offload the storage and provide a separate server to hand the SCADA data up to other systems in the ICS network. This allows the SCADA server to focus on its main job of collecting data and pushing control commands and lets the historian deal with all of the other systems that need this data.

So now that we've covered the central nervous system of a SCADA system, let's talk about all the external end-points. The goal of SCADA is to monitor and control the various devices that run our industrial processes. These devices can be just about anything you can imagine. For electric utilities, this is often various devices like sensors, relays, capacitor banks, feeder switches, actuators, and literally anything that a utility can think of to monitor the health of the power grid and control which homes/businesses are connected to which power sources. These various devices can have input/output requirements as simple as a single digital on/off relay or as complex as real-time sensors that communicate

on a specialized process bus. Because of the variety of I/O requirements, SCADA relies on intermediate devices to communicate with the disparate end-points. These intermediate devices are usually remote terminal units (RTU) or programmable logic controllers (PLC). Once again, there is a difference between these two devices, but that difference is fading over time and isn't important for this basic overview. These RTUs and PLCs are the devices that play the gateway and intermediary between the SCADA server and the end-points. Notice I said gateway AND intermediary. RTUs and PLCs do act as a gateway in the traditional IT sense (router and protocol/address translation), but these devices are also programmed with logic to make their own basic decisions based on the data that they are seeing from their I/O ports.

So, let's look at the whole picture. When an operator clicks a button in an HMI to send a control signal, this signal is sent to the SCADA server. This server sends the appropriate signal to the correct RTU or PLC. The RTU or PLC consults its pre-programmed logic to determine what it should do with this control signal and initiates the appropriate I/O responses on its attached end-points. If these end-points change the state of the process, this is usually picked up by process sensors, that send their data to their respective RTU or PLC which in turn sends this data back to the SCADA server. The SCADA server sends a copy of this new data to the HMI for the operator to see, while also sending a copy of this new data to the historian for record keeping and dissemination out to other systems that need this data.

I hope that helps to paint a clearer picture of SCADA systems for you readers.

### 3. What are the most prevalent Operating Systems seen in SCADA devices? Are there any proprietary OSes?

When it comes to the HMI, the SCADA server, and the historian, most modern systems being sold today are running on Windows, Linux or Unix. RTUs and PLCs are usually embedded electronic systems running some microprocessor controlled program or VxWorks, but many modern day RTUs being sold today are now running embedded versions of Linux. These are the general circumstances that I've seen, but realize that SCADA systems are broad and all encompassing. Anything is possible in SCADA. One-off solutions in some markets are very common especially historically.

### 4. What devices are network connected and which are not? What type of network is used: internal only, private with external access, the Internet?

In a generic sense, pretty much everything I discussed above is "network" connected. However if you are speaking about the more IT definition of "network" as in TCP/IP, pretty much everything is down to the RTU and PLC. Older RTUs and PLCs, which probably make up the majority of devices currently in use across the US, generally use serial links via dial-up modems or ISDN lines. However these have begun the slow transition to traditional TCP/IP in the last 10 years and will continue to do so.

Between the RTU / PLC and the end-points, this is still predominantly serial and parallel communications although sometimes standard-based and proprietary. Some of the modern "Smart Grid" devices being sold in recent years are using new high-speed process bus technologies which occasionally use lightweight protocols directly riding atop Ethernet, but this is not widely deployed yet.

As for public vs. private, the general rule of thumb is to deploy all SCADA devices on private networks with no direct links to the Internet. However all of your readers have probably seen media articles where some companies don't follow this best practice and get burned.

### 5. What are the most common attacks and on which devices?

It's definitely safe to say that the most common attack is on the commodity operating systems running the HMI, SCADA, or historian applications. As a penetration tester, the easier path is always from the corporate network or remote access VPN, through whatever services they permit through their ICS firewall to the Windows or Linux machine running in the ICS network. Once you gain control over one ICS machine, one simply pivots until they find gold. In this case it is usually the HMI application, as they are often point and click. To be honest, it isn't much different than doing a PCI pentest and trying to find the credit card data.

As for other attack surfaces, if you can gain network access to the RTUs and PLCs, these systems often run insecure

services like telnet, FTP, and TFTP. In some cases, passwords may not even be permitted on some of these interfaces, especially in older devices. Many modern day RTUs even run web interfaces, and like most web applications, they often sport a handful of security flaws. So once again, there isn't a lot different here for the traditional IT professional, however the one thing that is different is the sensitivity of these devices. You've probably heard the horror stories about how sensitive these devices are, and in many cases they are being knocked over by simple nmap scans. A researcher a few weeks ago commented that an nmap scan with OS versioning turned on crashed the PLC he was testing, taking him almost two full days to get it running again.

And finally, how easy is it for an attacker to cause something to happen in a SCADA network once he has gotten into the device? Well, it's easy to cause "something" to happen, however it is extremely difficult to cause "something you intend" to happen. This is because each ICS system (from an overarching system perspective) is custom designed. Think of it this way. If you were to number every light switch and power outlet in your house, and put them all on an interface with clickable buttons, how would you know what number to click to turn off your server rack (yes, I have one of those in my house&hellip;)? Perhaps you'd name it something instead of having just a number, but what if your naming scheme was limited to the good old DOS eight character naming convention? Remember, you might have 100 individual outlets in your house and at least 30 light switches. Well, I'm sure you'd figure out a good naming scheme, but that naming scheme probably wouldn't be the same as your neighbors. Now grow this scenario to the scale of an electric utility company. Now consider how complex these systems are when they don't just cover a single city, but multiple states and hundreds of cities.

Oh, and I forgot to mention that all of those end-points (switches and outlets in your house example) aren't just simple on/off controls, but complex systems that have RTUs and PLCs that have their own logic and make decisions based on the requests you make. Fortunately for us, most electric utilities don't have an HMI that has a clickable button labeled "nuclear reaction". Even something as simple as "kill power to Chicago" is not necessarily an easy matter even after finding the appropriate HMIs. Context is everything, and, without it, attackers are extremely limited in the actions they can cause to happen. Given enough time, attackers can gain that context, but it is not an easy matter. You can randomly flip switches, which is bad, but it isn't worse-case scenario.

6. Even without specific equipment named Human-Machine Interfaces, Social Engineering must be part of the attack surface. Can you talk a little about SE attacks and defenses?

Yes, social engineering is just as effective in ICS as it is in any other IT field. The benefit that ICS has though is the smaller numbers of persons that can effect change. Trying to social engineer your way into an ICS control center is equivalent to trying to social engineer your way into a SOC. Definitely possible, but it's about equivalent to trying to get to the random company's crown jewels.

7. What are the top 5 measures an organization can put in place to protect these critical systems?

They really aren't any different that any other IT organization. I personally like the concision of the SANS 20 Critical Security Controls. And while many of you may think I'm crazy, I also really like adapting PCI to whatever environment I'm trying to protect (swap PCI data for ICS data). However for a more exhaustive list of security protections, you may want to check out the NIST Interagency Report 7628 (700+ pages over 3 volumes) or the various security profiles released by ASAP-SG. For full disclosure, I should note that I played key parts in the creation of all of these documents, however these document are the primary ones used by the electric utility sector.

8. What unique challenges would a security professional face when performing a penetration test on SCADA systems in general as well as those associated with our nation's Critical Infrastructure?

I believe I've covered most of the issues above, however I should mention two other things. Number one is trust. It will not be easy to gain a company's trust to let you test their control systems. And number two is production vs. staging/testing. Because of the negative real-world effects your test can cause and the sensitivity of older SCADA devices, the risk is simply too great to test on production networks. In cases like this in the IT world, we simply setup a testing or staging environment, verify it is configured identically to the production environment and perform our test. However if you've been frustrated with the infrequency that traditional IT companies have in staging/testing their own environments, you'd be infuriated with the infrequency that industrial companies have in utilizing these testing environments.

If you want a more in-depth look at the methodology we use here at UtiliSec, please check out my upcoming talk and accompanying white paper at Black Hat Europe this coming March, Dissecting Smart Meters. If you simply can't wait, contact me, and I can share a prerelease copy of the whitepaper. You should be able to find me on Twitter, Facebook, or LinkedIn. You can also reach me directly by email at [justin at utilisec dot com].

Donald C. Donzal

Editor-In-Chief

The Ethical Hacker Network