Silver Bullet Talks with Ralph Langner

Interview

Gary McGraw

Cigital

Ralph Langner is the founder and CEO of Langner Communications, a German company focused on control system security. He has more than 20 years’ experience working with computerized control systems and was the first researcher to determine that Stuxnet was a directed cyberattack against Iran.

Hear the full podcast at www.computer.org/security/podcasts or www.cigital.com/silverbullet.

Gary McGraw: I first met you at Joe Weiss’s ACS Control Solutions Cyber Security Conference, where it seemed like I was one of a few security guys surrounded by a whole bunch of control systems people. Just how little do most control systems people know about cybersecurity?

Ralph Langner: There are a bunch of control system experts who really do know a lot of IT security, but, unfortunately, way too few. Engineers in the field are not as experienced as we would like because they just don’t have the time and budget. In most operations, control systems are more a matter for electricians than IT staff. This is no longer always true, but it takes a long time to change the culture. When we [first meet] the average client, we see very few engineers who have any experience in IT security. This has to change, especially after Stuxnet.

McGraw: I totally agree, but I will say that the quality of the engineers themselves [at the conference] was extremely high. The issue is just one of exposure to the whole notion of cybersecurity. There’s an important philosophical difference between trying to prepare for some sort of a reliability issue that may involve, say, random sunspot activity, and trying to prepare for intentionally malicious activity by an informed adversary.

Langner: One important problem that we have is that you can’t apply proven concepts from IT security to the plant floor. Just to give an example, you can look at the average controller, and you won’t find authentication/authorization—there’s no antivirus software on that controller, there’s no hard disk on it, there’s no patch management. But most of our vulnerabilities aren’t bugs as you know them from the IT world. The vulnerabilities that we’re most concerned with are actually legitimate product features, which is a very difficult problem because they can’t be patched away overnight, [due to] version conflicts. Many of our nastiest vulnerabilities will stay with us for several more years.

McGraw: I’m in total agreement with that, and most of the listeners to this podcast are as well, because our focus is much more in line with security engineering and software security—designing and building things properly—that it is with this sort of antivirus/firewall approach to IT security, which has failed us, frankly. Stuxnet is actually useful in getting people to understand the nature of the problem, possibly for the first time, even in the IT security domain.

What got you started analyzing the Stuxnet code in the first place?

Langner: As with anybody associated with security, we heard about Stuxnet last summer, but we didn’t take it seriously at first—to be absolutely honest—because so many aspects just didn’t make sense. For example, when it was speculated that the vulnerability for the SCADA database could probably be used by Stuxnet to exfiltrate intellectual property, it didn’t make sense. The SCADA database, that’s all raw data, so…

McGraw: It’s not very interesting, is it?

Langner: Right. Even if it had
Ralph Langner is the founder and CEO of Langner Communications, a German company focused on control system security. He has more than 20 years’ experience working with computerized control systems and was the first researcher to determine that Stuxnet was a directed cyberattack against Iran. Langner lives in Hamburg, Germany. Contact him at info@langner.com.

About Ralph Langner

Langner: It was absolutely amazing because we saw it already had meat, but it was still sniffing. It was checking out the target and obviously didn’t like it, which was one of the first clues that it was a directed attack because it wanted something very specific. By going into the debugger and configuring the PLCs in a certain manner that Stuxnet seemed to like, we were able to get more information and ultimately determine that it was 100 percent directed. It wasn’t just looking for a specific controller type—it was checking to see if the target configuration that would drive the centrifuge in Natanz was actually loaded on the controllers.

Langner: We threw some nice Siemens controllers at the beast, but it just didn’t want to eat.

McGraw: I understand that a typical configuration, especially in pipelines but also in centrifuge arrays, is to have a 417 control the 315s. Can you explain the DLL interpositioning that gets to the controller and then the modification of OB1 and OB35?

Langner: The DLL hijacked by Stuxnet is the link to the controllers. Again, controllers aren’t computers. They’re small, gray boxes; embedded real-time systems with no hard disk, no keyboard, no screen attached. These systems are programmed on a PC, usually on a notebook system, and that program, which we call ladder logic, is then loaded onto the controller, maybe via Ethernet, a proprietary Siemens connection, or a fieldbus link. To access any of these three, Siemens products use one driver DLL, which was hijacked by Stuxnet so that it could launch a man-in-the-middle attack and insert its own code loaded on the PLC.

Langner: That surprised me when we were debugging the code. But just to stay on this topic for one second, there’s another interesting point related to that original Siemens DLL. Even though the symbolic information was still in it, the average attacker wouldn’t know all the calling conventions. This is very difficult to reverse engineer, so it’s one more piece of

Langner: When one of their guys contacted me. We chatted very briefly, and he asked if we would be interested in cooperating. I said, “Certainly, yes,” but nobody followed up on the offer.

Langner: We’re completely independent. This research hasn’t been funded by anybody, and we’ve had very brief interactions with Siemens. The only request that Siemens had with respect to potential cooperation was at WeissCon, when one of their guys contacted me. We chatted very briefly, and he asked if we would be interested in cooperating. I said, “Certainly, yes,” but nobody followed up on the offer.

Langner: It was absolutely amazing because we saw it already had meat, but it was still sniffing. It was checking out the target and obviously didn’t like it, which was one of the first clues that it was a directed attack because it wanted something very specific. By going into the debugger and configuring the PLCs in a certain manner that Stuxnet seemed to like, we were able to get more information and ultimately determine that it was 100 percent directed. It wasn’t just looking for a specific controller type—it was checking to see if the target configuration that would drive the centrifuge in Natanz was actually loaded on the controllers.

Langner: We threw some nice Siemens controllers at the beast, but it just didn’t want to eat.

McGraw: I understood that a typical configuration, especially in pipelines but also in centrifuge arrays, is to have a 417 control the 315s. Can you explain the DLL interpositioning that gets to the controller and then the modification of OB1 and OB35?

Langner: The DLL hijacked by Stuxnet is the link to the controllers. Again, controllers aren’t computers. They’re small, gray boxes; embedded real-time systems with no hard disk, no keyboard, no screen attached. These systems are programmed on a PC, usually on a notebook system, and that program, which we call ladder logic, is then loaded onto the controller, maybe via Ethernet, a proprietary Siemens connection, or a fieldbus link. To access any of these three, Siemens products use one driver DLL, which was hijacked by Stuxnet so that it could launch a man-in-the-middle attack and insert its own code loaded on the PLC.

Langner: That surprised me when we were debugging the code. But just to stay on this topic for one second, there’s another interesting point related to that original Siemens DLL. Even though the symbolic information was still in it, the average attacker wouldn’t know all the calling conventions. This is very difficult to reverse engineer, so it’s one more piece of

Langner: We’re completely independent. This research hasn’t been funded by anybody, and we’ve had very brief interactions with Siemens. The only request that Siemens had with respect to potential cooperation was at WeissCon, when one of their guys contacted me. We chatted very briefly, and he asked if we would be interested in cooperating. I said, “Certainly, yes,” but nobody followed up on the offer.

Langner: We threw some nice Siemens controllers at the beast, but it just didn’t want to eat.
evidence that the attackers must have had insider knowledge about the Siemens product. Several other areas indicate a high level of intimate knowledge with the Siemens architecture and product, indicating that this can’t have been reengineered by, say, your most skilled hacker in a reasonable amount of time, let’s say one or two years. It just isn’t possible.

**McGraw:** It takes more expertise than that.

**Langner:** We ended up with a hijacked legitimate driver DLL. The code goes on the controller and loads a lot of stuff on it, so in a sense, this stealth control system now runs concurrently with the legitimate code. The legitimate code is still on the controller, and it’s executed, but there’s this stealthy part, the rogue part, that monitors operation of the I/Os, and that’s what…

**McGraw:** That’s what makes it kind of like a rootkit for the controller. I think it’s also important to understand that the payload itself makes use of that original controller code that’s sitting around.

**Langner:** I don’t like the term rootkit in relation to controllers because the architecture of these controllers is so different from what you see on a PC or on, say, a Linux box that the term really doesn’t apply well. We do have some stealthiness here, and we do have some advanced attack technology, but it’s really quite different from what we know and see in the IT world.

**McGraw:** Here’s why I think it’s like a rootkit, even though I’m fine with not using that term. The notion is of standing in front of an I/O stream, so that when you get a control call or a query—say, about what’s in that register—you control what you say about what’s in that register and are thus able to hide from simple queries that are looking for the infection.

**Langner:** Well, that’s correct—it’s quite stealthy. On the other hand, in certain other parts, it’s not stealthy at all. For example, just to hijack and rename the original DLL, you do that once. After Stuxnet, this technology won’t work because it’s just too easy. Certain other things apply to the controller attack, too—for example, you just have to use a version control system that doesn’t use the infected vendor’s driver, and you would see the changes immediately. It’s just for as long as you stay in that basic average engineering environment that the attack would remain stealthy. It’s not rocket science; it’s much more basic than your average Windows rootkit.

**McGraw:** It’s a target that hasn’t been hit before, so it hasn’t been hardened and is susceptible to pretty basic attacks.

**Langner:** That’s absolutely right. I’ve heard hackers say, “Yeah, Stuxnet, this associated code, we could have done it so much better.” But it didn’t need to do it any better because the attackers had the advantage of surprise. Nobody was expecting it.

**McGraw:** These guys knew about the Siemens systems, and these hacker boys just don’t. It’s that simple.

**Langner:** Certainly there are several areas where Stuxnet can be improved. We think it will be improved for version 2.0, but it just wasn’t needed this time.

**McGraw:** Can you explain the payload in a little bit more detail and get into the OB1 and OB35 modifications?

**Langner:** As I said, the rogue DLL infects the controllers. It injects what I call a stealth control system that runs in parallel with the legitimate code. The important point here for the attack is that you must get this additional rogue code called, and as was seen in Stuxnet, this is actually very, very easy because all you need to do is insert calls to your rogue code at the beginning of the two code blocks that you mentioned, OB1 and OB35.

In the Siemens world, OB1 is the main routine that’s called every time you start up the controller. For C programmers, it would be your main function, but the difference is that this function is called in a loop, so it’s called over and over again, many, many, many times per second, meaning that once you manage to get in at the beginning of OB1 or any other code block, you’re in business. This is what Stuxnet did: it inserted function codes at the beginning of this code block, and depending on the return codes, it decided whether to pass control along to the legitimate code.

OB35 is a 100-millisecond timer that you find in every Siemens 7 PLC. It’s an interrupt handler or event handler that’s called automatically by the operating system of the controller 10 times per second. Once you insert your rogue code there—again, at the beginning of the block—it will be called 10 times per second. The problem is how easy it is, in technical terms, to actually do that, to make that happen. There is no check-in, so the controller does not do any checks. If this is really authentic and code integrity isn’t violated…

**McGraw:** Then it just does it.

**Langner:** You just insert your rogue code into the code for the legitimate OB1, and you’re in business. This is a big, big problem
for us because it actually works on any Siemens S7 around the world, and there are several millions of these controllers.

**McGraw:** I was going to ask you about how many Siemens control installations are vulnerable to similar attacks. Thousands?

**Langner:** No, millions.

**McGraw:** A lot of people are familiar with writing graphical user interface code or at least some sort of an event handler that does user interface stuff. This is kind of like rendering, where the notion is that you flush the buffer. It runs once every few clicks regardless, so it’s the control loop that you’re getting into.

**Langner:** Yes, that’s the basic attack routine and/or attack technology. As I said, it’s a very big problem because it works with every Siemens controller, and you find these controllers in power plants, in chemical plants, in food and beverage companies—they aren’t application specific. They aren’t only used in uranium enrichment facilities.

**McGraw:** Beyond the OB1 and the OB35, what would you call them? Frequency converter attacks go even further than just those that get down into the specifics of the Natanz centrifuge facility.

**Langner:** Yeah, but the important distinction here is that what Stuxnet does against the frequency converters is application specific; code injection into OB1 is not—it’s generic. The frequency converter manipulations, on the other hand, will only work with specific models and installations, so it’s not that much of a problem.

**McGraw:** I remember there was something about querying how many 315s were attached to a 417. It was something between 31 and 2,000. It had to be a specific number.

**Langner:** No, that’s a little different. The attack code actually queries how many frequency converters are attached to the 315. The thing with the 417, that’s a different story, but to stay with the 315 just for a second here, there was another very nasty generic attack technology, which is overriding a system function of the operating system or library function. It would be similar to—let’s say, again, as in C programming, which I always use because it’s the last program language I used in IT—doing something like overriding your printf function call. So, every time you call printf in your application, well, it does print something, but it does something in addition that you don’t recognize, and this is another attack technology that you see on 315. The code overrides the system function call for reading values from the peripheral—in this case, the frequency converters—so it gets an understanding of how many of these frequency converters are attached, how fast they’re running, et cetera, et cetera. Again, this is generic. As with the problem I mentioned earlier, it’s big because this attack will also work with valves, pumps, et cetera. It’s not tied to a specific application.

**McGraw:** So, could you change that 1-KHz filter into whatever value you wanted it to be?

**Langner:** Yeah. Stuxnet doesn’t do that, but again, the funny thing here is that it tells us a lot about the background of the attackers, because very few people even knew that this would be possible. When you program controllers, you take the system functions for granted. You would never think about touching the system function, and the attackers knew, well, they can be overwritten and the program would still work.

**McGraw:** It seems obvious to me that Natanz, the Iranian centrifuge facility, is the real target, and maybe Bushehr, too, but in the beginning, most of the coverage was about Bushehr and not Natanz.

**Langner:** Yeah, that’s my fault.

**McGraw:** Thanks a lot, Ralph.

**Langner:** Well, this is actually quite easy to explain. First, Bushehr is not Stuxnet’s target. That’s 100 percent certain right now because we’ve been able to link the data structures that we have in the code to the actual plant layout in Natanz. That was very important. We discovered this in late December—probably 27 December.

**McGraw:** It seemed obvious to me, even in Maryland, but it was surprising that the Symantec guys were still talking about the delivery mechanism and not the payload.

**Langner:** Yes. Well, but they are not control system experts, and if you really want to determine what the target is, you can’t remain in code analysis, you have to establish links to the outside world. To do that, you must be able to examine, for example, how centrifuges work, how a power plant works. My [mistake] was back in September, when I made the connection to the Iranian nuclear program, that I identified Bushehr as the most likely target.

**McGraw:** Just because it’s a nuclear power plant.

**Langner:** But this was wrong, as we know now, but it was just from my basic layman’s understanding of where the significant strategic targets were. So, for example, I knew that Israel had attacked the
nuclear power plant in Iraq and it had attacked the nuclear power plant in Syria, so to me, it looked like a natural target. Everything we researched about the control systems in a nuclear power plant, especially in Bushehr, contributed to that theory, so it was strange that we had been able to nail down that the Siemens 417 controller is actually used in Bushehr. This just kept us on the wrong path—the major breakthrough came on 27 December when I learned how a centrifuge cascade is structured. We did some additional background research and got a fairly good understanding of the organization.

**McGraw:** For those people who aren’t so familiar with Siemens systems, 417s are expensive, and there usually aren’t very many of them, but 315s, especially 315-2s, are cheap, so there tend to be lots of those.

**Langner:** That’s correct. The 417 is the top-of-line model of the Siemens controllers. You don’t need it to run a cookie plant, for example, so it’s used for big stuff, for complex plants. From what we understand, the 417 is especially used for safety in Natanz. The big difference with your basic production system and a safety system is that while the production system produces, for example, uranium, the safety system is in place to make sure that the whole joint doesn’t blow up when something goes wrong.

**McGraw:** I think the same design is used in pipeline controllers.

**Langner:** Yeah, that’s the same design you would find in a chemical plant, for example—when you must make sure that certain thresholds in terms of pressure or temperature aren’t exceeded. There are several misconceptions about the safety issue—it’s not about fooling operators sitting in a control room. What we see in Stuxnet is much worse. The Stuxnet attack on the 417 actually fools the automated safety systems. Usually in safety, you have several different layers, with the most important—the automated layers—at the front. Your operator is actually the last line of defense. The front lines are very, very important because this is where you must react within a split second.

**McGraw:** And it happens at superhuman speed, so you have to have the controller notice.

**Langner:** Absolutely, and you could assume that the operator in the control room just fell asleep or is out taking a leak. But, for any installation where really bad things can happen, you are definitely interested in having a good automated safety system.

**McGraw:** I want to shift gears a bit and talk about some of the meta aspects of this work. You’ve garnered a ton of press, and I wanted to ask you two quick questions. What’s the best coverage so far, in your view, and who wrote the silliest, most ridiculous thing?

**Langner:** One of the best pieces was certainly done by *The New York Times*. You can’t imagine the effort that *The New York Times* team put into that short story. I’ve heard many complaints, “They aren’t giving any facts or evidence,” but as one of their prime sources, there are many, many pieces of evidence that they did not tell. I think they will at some point in time. Obviously, some additional research and reporting is going on, but that was really, in my view, the highlight. For the worst article as well, I don’t want to blame anybody here.

**McGraw:** Oh, come on.

**Langner:** I’ve learned that there is a huge difference between good and bad journalism, but I would like to point out that the media in total did an outstanding job. I would like to point out that for a specific reason, because there are other organizations who actually are obliged to inform you and everybody else what Stuxnet is all about…

**McGraw:** And they didn’t do it.

**Langner:** They are paid with taxpayers’ money, and they didn’t do it.

**McGraw:** Well, they don’t understand it.

**Langner:** I wouldn’t say so. For example, if you look at DHS, it certainly has the experts. It has many more resources than we have, but it doesn’t tell you what Stuxnet is all about. It doesn’t deliver research results.

**McGraw:** Well, you’re giving DHS more credit for understanding cybersecurity than I think it actually should have, frankly.

**Langner:** I can tell you one thing
**McGraw:** To your credit, it’s very important that you covered this the way that you did, because there were so few people with technical expertise in control systems who were involved in the early days. This story could’ve spun off into yet another stupid virus 0-day blah, blah, blah, all the usual people banging the usual drums about cybersecurity and never even got into the control systems part of it, so thanks for that. We appreciate it.

**Langner:** Well, thank you.

**McGraw:** What do you think we should do to educate control systems engineers about security?

**Langner:** We have to do a lot, and quick, and there are so many places where we could start. That’s not the issue. The major point is to get things started and to get the budget, to get the funding for this, and if you talk to the average control system engineer, you’re running into open doors. These guys understand the problem. What we have to do is to convince the CEOs. They haven’t gotten it so far because they only understand, well, this is going to cost a lot of money, and, well, we haven’t seen many control system security incidents before.

**McGraw:** Even the Siemens guys are that way. I mean, they’ve barely started with their software security program, and I’ve been talking to them about it for more than a year now.

**Langner:** Absolutely. You don’t need to convince the engineers. They got the message, but they don’t have the budget. And they don’t have the time.

**McGraw:** Well, that sounds like a standard problem in computer security risk. I have one last question that has nothing to do with Stuxnet at all. What is your favorite piece of recent music?

**Langner:** I’m an old guy, so I don’t listen to top of the pops, but one of my favorite records is by Marc Anthony, *Valio La Pena*. I like Salsa. I like Latin music. I like Marc Anthony very much.

Show links, notes, and an online discussion can be found on the Silver Bullet webpage at www.computer.org/silverbullet or www.cigital.com/silverbullet.

---

**Gary McGraw** is Cigital’s chief technology officer. He’s the author of Exploiting Online Games (Addison-Wesley, 2007), Software Security: Building Security In (Addison-Wesley, 2006), and seven other books. McGraw has a BA in philosophy from the University of Virginia and a dual PhD in computer science and cognitive science from Indiana University. Contact him at gem@cigital.com.

---

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.