HOW STUXNET DEMONSTRATES THAT SOFTWARE ASSURANCE = MISSION ASSURANCE

HP Enterprise Security Business Whitepaper

For decades, the people responsible for guarding most U.S. critical infrastructure facilities have felt that cyber security just doesn't apply to them. Their facilities rely on industrial control systems (ICS)—custom-built suites of systems that control essential mechanical functions of power grids, water treatment plants, and the like—usually not connected to the Internet, also known as "air-gapped." Many industry owners, operators, and regulators believed that this security model provided an infallible, invulnerable barrier to malicious cyber attacks from criminals and advanced persistent threat (APT) adversaries alike.

Until 2009, they were pretty much right. Until Stuxnet changed the rules of the game and revealed just how vulnerable they are.

ENTERPRISE SECURITY

Discovered in July 2009, the Stuxnet family of malware can enslave, damage, or even destroy industrial control systems and the vital equipment they run. Initially inserted via flash drive (thus bypassing the air gap), the malware exploits previously undiscovered (zero-day) vulnerabilities in Microsoft® Windows® applications. Despite infecting thousands of facilities worldwide, the worm has damaged none except those of its apparent target: Iran's nuclear program, which has experienced severe unexplained delays—since mid-2009.

Multiple investigations by top-tier news organizations and think tanks have since revealed Stuxnet's devilishly ingenious design. Its two-pronged attack includes sabotage and subterfuge: Spin nuclear centrifuges so fast they self-destruct, while preventing plant operator interference by showing normal-looking, but fake, data on monitoring instruments.

Stuxnet's sophistication lies in its exquisite tailoring. It attacks not just specific makes and models of extremely specialized equipment, but also exact configurations that perform processes unique to individual facilities. For instance, Stuxnet seeks a series of 984 linked machines—the same number of centrifuges the Iranian uranium enrichment site at Natanz took offline in mid- to late 2009, an independent report later found.

Stuxnet infected not only Iran's nuclear enrichment facilities but at least one of its Russian-built nuclear reactors. Invoking nightmarish memories of the legendary 1986 nuclear disaster in the then-Soviet Union, Dmitry Rogozin, Russia's ambassador to NATO, told Reuters in January 2011 that Stuxnet had infected systems at Iran's Bushehr reactor and "could lead to a new Chernobyl." This and other reactions to Stuxnet underscore how, more than any previous cyber incident, a digital attack can have potentially devastating physical consequences.

Stuxnet brings new meaning to the term "weapon of mass destruction."

Software security assurance: no longer just a good idea

Stuxnet changed the game of cyber security by shattering assumptions long cherished by many in the public and private sectors: Organizations not connected to the Internet or expecting their network defenses to protect them have negligible risk of cyber attack and do not need to eliminate vulnerabilities in software applications. Security through obscurity is no longer a viable defense, as adversaries can know more about target systems than system owners do. Organizations must now act quickly, as Stuxnet provides an attack blueprint anyone can use. The consequences of inaction could be dire.

Fortunately, both awareness of the problem and the will to solve it have never been higher. What many organizations still lack, though, is specific knowledge of what to do next. They need expert insight to create a comprehensive software security assurance (SSA) strategy.

SSA is a set of best practices and technology that helps organizations reduce security risk in software by removing exploitable vulnerabilities from application source code and preventing their introduction in the development process. Critical infrastructure—not to mention the federal government—uses predominantly custom-built systems that are often hybrids of new and legacy technology. Many of these systems were built with features and performance in mind, not security, and thus they are riddled with vulnerable applications. In the age of Stuxnet, software assurance is mission assurance.

Many government and industry leaders in the software security community have advocated for years for widespread SSA implementation. They have struggled with insufficient official guidance, the absence of dedicated funding sources and lopsided emphasis on hardware and network security instead of software, where industry analysts believe more than 75 percent of attacks occur.

Fortunately, both awareness of the problem and the will to solve it have never been higher. What many organizations still lack, though, is specific knowledge of what to do next. They need expert insight to create a comprehensive software security assurance (SSA) strategy.

Stuxnet and other major application attacks in recent years have prompted changes in federal law to solve these problems and make software security assurance a top priority. This year's National Defense Authorization Act (NDAA) (HR 6523), includes a section titled, "Strategy on Computer Software Assurance." The 2011 NDAA was the first law to provide strong policy guidance to secure both new and legacy software from attack throughout the software development lifecycle. "To program managers it is a wake-up call to transform their internal processes across the lifecycle of software development and implementation," says Robert Lentz, president of Cyber Security Strategies. "I am very optimistic that this could be the tipping point we have long sought."

Until his retirement in October of 2010, Lentz was a Deputy Assistant Secretary of Defense and the Pentagon's point man for cyber security and information assurance, and had spent his entire 35-year career at the National Security Agency (NSA) and in the Defense Department. He believes that the new SSA legislation "not only shows the high interest by Congress in this topic but sends a dramatic signal to the national security community (especially leadership) that protecting software is a major priority."

The new guidance regarding SSA strategies will significantly affect software development practices for agencies, contractors, and systems integrators seeking DoD software certification. Other federal agencies and large commercial enterprises could follow DoD's example and mandate software security assurance in their own policies and acquisitions. This important new law could therefore inspire needed and better security across industry sectors.

What organizations can do now to help prevent a "son of Stuxnet" in the U.S.

Stuxnet has severe ramifications for the federal government, even though it did not specifically target the United States. Although roughly 85 percent of critical infrastructure is privately owned and operated, the federal government is partly responsible for helping protect it. Experts note that Stuxnet's success and notoriety will likely spawn imitators, some of whom will attack the U.S. government. Since the report of Stuxnet, we have learned of Night Dragon, a sophisticated breach of energy companies, RSA's SecureID breach, and Comodo's certificate hack (attributed to Iran), the Lizamoon mass SQL Injection heist, and the huge Epsilon email breach. What do these hacks have in common? They all had network security controls, and yet their software was breached. The problem is large and complex, and only getting more so.

Fortunately, there are steps organizations can take now to support software security assurance:

From the "7 Practical Steps to Delivering More Secure Software" white paper

1. Quickly evaluate the current state of software security and create a plan for dealing with it throughout the development lifecycle.

2. Specify the risks and threats to the software so they can be eliminated before they are deployed.

3. Review the code for security vulnerabilities introduced during development.

4. Test and verify the code for vulnerabilities.

5. Build a gate to prevent applications with vulnerabilities from going into production.

6. Measure the success of the security plan so that the process can be continually improved.

7. Educate stakeholders about security so they can implement the security plan.

Any development organization can implement this security plan immediately and begin to receive a return on its efforts within a minimal period of time. The key is to start now.

To complement the software strategy, there are several other areas of good security practices to observe and implement if they are not already part of the organizational security approach:

- Implementing software configurations such as the U.S. Government Configuration Baseline (formerly the Federal Desktop Core Configuration), strong authentication, and strict, documented internal policies and procedures
- Asking vendors to provide guarantees of software security as required by HR 6523
- Inserting and enforcing software assurance requirements in contracts
- Reviewing IT security policies to ensure all users of organizational networks and data comply with the strictest security policies possible with respect to the mission

Finally, organizations should determine how much they can spend, how much risk they can afford, and who is accountable for that risk. Constructing a new building in parts of California without accounting for earthquakes is unacceptable. Thanks to Stuxnet, building software without accounting for security is no longer an acceptable risk.

Rob Roy is the Federal CTO of Fortify, an HP Company, and a major provider of software security assurance products and services to the federal government.