U.S. Resilience Project

CASE STUDY

Telvent Worldwide
Based on an interview with
Jeff Meyers, Director for Smart Grid Sales, Telvent

February 7, 2012

Securing Information on the Smart Grid: Telvent Supply Chain Best Practices

Telvent is an information technology company that specializes in real-time data collection and monitoring systems and operational tools to transform data into actionable information. One of its key business areas is smart grid applications and tools. Telvent applications are in use in engineering and operations departments at more than 550 utilities in North America and around the world. From core geospatial network modeling and management, to real-time analytics and control, Telvent builds software to enable the smart grid.

Best Practices in Software Development

Telvent develops software and protects it through a common set of security practices that are appropriate and consistently implemented, and people who are vetted for capability and experience, as well as a potential for malicious intent.

Across its diverse product lines — from GIS systems to the outage models — Telvent's programmers manage approximately 3-3.5 million lines of code.

Telvent uses Agile software development, a methodology based on iterative and incremental development and collaboration between cross-functional teams. The Agile approach offers competitive advantages in terms of adaptive planning and flexible response to change, but it has some built-in security safeguards as well.

Coders work in pairs for actual programming tasks. On the surface, any attempt to build disruptive or malicious functionality (malware) into the code would require at least two people working in tandem. In fact, even the coding pairs could not succeed in delivering code with embedded malware. The methodology dictates that teams never build anything that takes longer than two and a half weeks (a "sprint"), which could be anything from a couple of hundred to a couple of thousand lines of code. Each sprint involves at least one code review, during which members of the team "walk through" each other's code. Functionality is tested at the end of each sprint against vetted requirements by a QA specialist assigned to the team. To introduce malware into an application in an Agile system would likely require the complicity of everyone on the subteam, approximately four to eight members, including the product owner, a senior programmer with both management and coding skills.

Best Practices in Software Testing

A second level of security is attained during the testing process. Every software development organization tests. At Telvent, however, this is not a separate activity after the product development is complete. Testing is built into the development process from requirements validation, to unit testing for each sprint, to production testing for each software release. Once during each release cycle, each project team takes a one-day break in the coding cycle to stress test. This exercise, called "SWAT" (Software With A lot of Testers), takes place at a known date prior to release and is an all-hands-on-deck exercise in which all programmers stop coding and start testing, looking not only for quality bugs but security issues: holes, places in the code with a single sign-on, hard-coded paths, legacy protocols, anything that creates or increases the threat surface. The rewards are geared toward finding and learning from mistakes, and there are prizes for those who find the most bugs and the most significant security threats.

Beyond human testing, Telvent uses machine-based automated testing scripts for highly complex scenario testing, as well as for regression testing. Automated testing is particularly valuable when used to evaluate the impact of newly released code on legacy applications. Machine-based testing can simulate multi-user conditions and highly repetitive tasks. While not specifically able to sniff for malware, automated test scripts can discover functional anomalies based on repetitive use conditions that can be base triggers for malware such as Trojan horses or other kinds of disruptive functions.

Best Practices in Software Design

Smart grid technology itself is often seen as a potential security problem because it opens utility grids to many potential penetration points, including the Internet. A smarter grid requires integration among systems that have traditionally been isolated, further extending the threat surface. But application of standardization and interoperability principles can increase the security of the smart grid. Standard architectural patterns and standard integration techniques make it possible to create great efficiencies, but also enable operators to identify anomalies more easily.

Telvent adheres to key architectural principles that enable the company to design in, rather than add on, security. By adopting a standard reference architecture, such as Microsoft's Smart Energy Reference Architecture, vendors can ensure that the integrated environment is built upon a foundation that has been designed with cybersecurity as a key requirement. Further, sticking to industry integration standards, such as the Common Information Model, allows for predictable integration with systems and devices beyond those delivered by a single vendor. Standard integration practices reduce customized code, a key failure point and a critical opportunity for cyber threat. Finally, solid architecture allows for the straightforward embedding of intrusion and malware detection and tamper-proofing tools that are built to provide internal security.

The most secure software products must eventually leave the development shop and be implemented in the real world of grid modernization. Implementation means that grid management software must touch and be touched by legacy systems and external devices with varying levels of security design and management tools. By adopting a standard architecture and using standard integration techniques, the threat surface from these external factors is significantly reduced.

Gaps and Ongoing Improvements

No software product or system is 100 percent foolproof, and even the best development methodologies have room for improvement. Among the most crucial issues and key areas of concentration for Telvent are:

- **Harmonization of methodology and security practices.** Most modern software is not built in a single physical location, and Telvent is no exception, with development teams in two North American and one European locations. Although each team uses its own consistent practices, harmonizing those across all teams would enhance overall security.
- **Securing implementations.** Grid management software must be implemented in the real grid. Implementation teams often consist of both vendor and utility staff with varying backgrounds, capabilities, and degrees of vetting. Internal utility IT teams may have existing practices or methods that must be harmonized with the vendor's to ensure consistency and close gaps in security.
- **Ongoing surveillance of implemented technology.** Delivering a system in a secure fashion does not guarantee that it will remain so indefinitely. Telvent uses strong-naming and code signing techniques to ensure tamper protection, but it and its clients could do more to ensure that patches, upgrades, and new integrations do not compromise system security.
- **Future deployments using cloud computing technology.** Most Telvent clients report that cloud computing is not currently an attractive option for mission-critical grid management applications. However, there is still work to be done to ensure that any cloud deployments that may touch or impact grid management tools are properly vetted, and that any future applications are designed with the rigor of system-based platforms.