Worlds in Collision-
Ethernet and the Factory Floor

Eric Byres, P.E.	Joel Carter	Amr Elramly	Dr. Dan Hoffman
Research Manager	Research Assistant	Research Associate	Associate Professor
BCIT Internet	BCIT Internet	BCIT Internet	University of Victoria
Engineering Lab	Engineering Lab	Engineering Lab	Dept. Computer Science
Burnaby BC, V5G 3H2	Burnaby BC, V5G 3H2	Burnaby BC, V5G 3H2	Victoria, BC, V8W 3P6
ebyres@bcit.ca	jcarter@bcit.ca	aelramley@bcit.ca	dhoffman@csr.csc.uvic.ca

KEYWORDS

Security, Ethernet, TCP/IP, Network Intrusion, Hacking, Control Systems, PLC, DCS

ABSTRACT

Over the past few years the world of industrial controls has borrowed substantially from the world of information systems. Technologies such as Ethernet and TCP/IP have made the interfacing of industrial equipment much easier, but there is now significantly less isolation from the outside world. Network security problems from the business network can be passed on to the process network, putting industrial production and human safety at risk. This paper evaluates these risks to industrial control systems from both accidental and malicious intrusion. The first portion summarizes an analysis of reported incidents in industrial environments and their effects on process systems. The second part describes a series of tests developed and conducted at the BCIT Internet Engineering Lab to determine possible security weaknesses in common programmable logic controllers. Based on these results, recommendations are presented on designing network security for critical industrial control installations.

INTRODUCTION

Over the past ten years industrial control systems have seen a significant increase in the use of computer networks and related Internet technologies to transfer information from the plant floor to supervisory and business computer systems. For example, most industrial plants now use networked process historian servers to allow business users to access real-time data from the distributed control systems (DCS) and programmable logic controllers (PLC). There are also many other possible business/process interfaces, such as using remote Windows sessions to the DCS, or direct file transfer from PLCs to spreadsheets. Regardless of the method, each involves a network connection between the process and the business systems.

At the same time, there has been an explosion in the use of Ethernet and TCP/IP in industry for process control networks. For many years the control systems used proprietary industrial networks, such as Data Highway Plus or Genius I/O, giving them a considerable degree of protection from the outside world. Today many DCS and PLC systems use protocols like Ethernet, TCP/IP, and HTTP as a critical component of their architecture, resulting in easier interfacing at the cost of less isolation and security.

CONFLICTING CULTURES

While technologies such as Ethernet and TCP/IP allow for significant cost savings and improved interfacing for industry, it is important to understand that their origins are rooted in a culture very different from the factory floor. Even the neophyte Internet user can spot these differences in terms of reliability – occasional failures are common and tolerated on the Internet while most control systems are expected to operate for months, if not years, without interruption. Similarly, the tradition of beta testing many new Internet products in the field and recovering from problems by simply rebooting servers or switches contrasts sharply with standard plant floor practices. This is not surprising since the risk impact of outages on the Internet are typically loss of data, while outages in the process environment will certainly result in loss of production and may even cause loss of equipment or life.

Very simply, the Internet culture and the technologies that it has created are based on the idea that performance is paramount and outages, while undesirable, are acceptable. This is clearly not true for the industrial system.

Nowhere is this cultural difference more pronounced than in the area of cyber security. Considerable media and research attention has focused on the topics of Internet viruses and hacking, but the reality is that most Internet hosts are only lightly secured. For example, KC Claffy of the Cooperative Association for Internet Data Analysis (CAIDA) reports that plaintext passwords are still very common on the Internet, a clear violation of the most rudimentary security standards (1). Similarly, until recently most IP networks were openly connected to the outside world, while the factory engineer has always demanded that the control system networks be isolated from the rest of the company information systems. Even where security is well defined, the primary goal in the Internet is to protect the central server and not the edge client. In process control the edge device, such as the PLC or smart drive controller, is considered far more important than a central host such as a data historian server.

Looking at these differences in needs and cultures, which the authors have attempted to summarize in Table 1, it is clear that the industrial control world must not blindly accept the solutions of the Internet world. These technologies may be extremely useful but they require careful consideration before they are implemented on the plant floor.

To understand how the solutions need modification in terms of cyber security, this paper starts by looking at a number of industrial cyber security case histories and the lessons these can teach us about the dangers of blind adoption of Internet security strategies on the plant floor. We then discuss the security analysis of a specific PLC and finally we close with a series of recommendations for a sound plant floor approach to incorporating Internet Technology.

TABLE 1: COMPARISON OF INTERNET AND FACTORY FLOOR EXPECTATIONS AND PRACTICES

	Internet	Factory Floor
Reliability	Occasional failures tolerated Beta test in the field acceptable	Outages intolerable Thorough QA testing expected
Risk Impact	Loss of data	Loss of production, equipment, life
Performance	High throughput demanded High delay and jitter accepted	Modest throughput acceptable High delay a serious concern
Risk Management	Recover by reboot Safety is a non-issue	Fault tolerance essential Explicit hazard analysis expected
Security	Most sites insecure Little separation between intranets on same site Focus is central server security	Tight physical security Isolated MIS network from plant network Focus is edge control device stability

REPORTED INCIDENTS

The British Columbia Institute of Technology Internet Engineering Lab (BCIT/IEL) maintains an industrial cyber security incident database that tracks incidents involving process control systems in all sectors of manufacturing. While most companies are reluctant to report cyber attacks or even internal accidents, there are now enough events to allow some basic statistical analysis of the data.

Since the initiation of the tracking project, 22 incidents have been logged in the database. The first conclusion we can draw is that there is a problem and it may be more widespread than most process engineers believe. Of these incidents, employees caused over 50% of them. This correlates well with data from FBI studies on the sources of cyber attacks:

> *A study by the FBI and the Computer Security Institute on Cybercrime, released in 2000 found that 71% of security breaches were carried out by insiders. This is supported by the realization that persons with high technical skill and process knowledge pose the greatest threat to an organization*. (2)

In other words, most of the security risks to a control network may not be an Internet teenager on a joy ride, but rather, a disgruntled employee. In fact, one of the first reported cases of plant floor “hacking” occurred in 1988 on an Allen-Bradley DH+, which was used by an angry worker to modify a different department's PLC-5. He changed the password to something obscene, blocking all maintenance access to the system (it was believed that he had found the original password on a post-it note).

Incidents like the one noted above also highlight the fact that the standard IT firewall approach to security will not protect the plant floor since most incidents are occurring inside the firewall. Even when the attackers are on the outside, firewalls don’t always protect the factory floor. For example, in October 2001, an Australian man was sent to prison for two years after he was found guilty of hacking into a waste management system and causing millions of liters of raw sewage to spill out into local parks, rivers and the grounds of a Hyatt Regency hotel (3). He did it because the area’s Council rejected the job

application he had made to work as a controls engineer at the plant. Court reports show that he attacked the control system not through the firewall, but through a wireless network used for SCADA control.

Often the incidents are not deliberately malicious but the results are devastating nonetheless. A good example of this type of problem occurred in a large East Coast paper mill in 1998 (4). The mill had just completed an upgrade of its paper machine, during which a number of engineers had been brought in from head office to assist with DCS commissioning. Everyone on the DCS commissioning team knew the passwords for the control system computers and when the project was completed, no one bothered to change them.

Trouble started about a month later when one of the head-office engineers decided he needed a good data source for an expert-systems experiment he was running. Using the company's wide area network (WAN), he was able to connect into the mill network from the corporate headquarters several hundred miles away. Once into the mill's business LAN, he was able to connect to the DCS through a link originally set-up to allow mill supervisors to view operators screens from their offices. He then loaded a small program onto one of the DCS graphics stations (a UNIX machine). This program asked all DCS devices to dump their data back to him once every five minutes.

All this would have worked fine, except that the engineer's new task would occasionally overload one of the DCS to PLC communications gateways, and it would stop reading the PLC data. This, of course, caused the machine operators great panic as they lost control of the motors controlled by the PLCs. Soon the electrical department was busy troubleshooting the PLCs. Meanwhile the head-office engineer had left the company to work for a competitor.

Eventually the problem was solved by an eagle-eyed mill engineer who noticed that the problems always occurred at intervals that were at multiple of five minutes. Suspecting that it might be software induced, he started to inspect all the tasks running on the DCS computers and found the offending task. Of course, by then the lost production in the mill had been substantial.

A CASE STUDY IN PLC SECURITY TESTING

From the above data, we have come to believe that it is naive to assume that control devices will never be exposed to some sort of internal or external cyber incident. PLCs and DCSs need to be hardened so that any intrusions that do occur will have little direct impact on the industrial process.

Are today’s controllers tough enough to withstand some level of network attack? To answer this question the BCIT/IEL decided to develop a series of test procedures to test industrial controllers for their susceptibility to various network problems, including standard hacking attacks. This section presents five tests for security, carried out on a popular PLC:

1. **Open ports**. Unnecessarily open TCP and UDP ports are a common security loophole. While at least one port must be open for normal PLC communications, it is important to ensure that (a) all open port(s) are well protected and (b) that no un-required ports are open.
2. **Simple Network Management Protocol (SNMP) robustness.** SNMP provides access to network devices, for monitoring and configuration control. SNMP security is weak, making it a significant security concern.

3. **Malformed packets**. Some TCP/IP implementations are vulnerable to attacks based on packets with purposely illegal field values.
4. **Broadcast traffic storms.** A broadcast message is a message that is directed to all hosts on a network or subnet. Unusually large numbers of broadcast messages can cause host failures.
5. **Resource starvation**. Many TCP/IP hosts are vulnerable to attacks based on consuming system resources to the point that normal operation ceases.

The equipment required for the tests included a widely used PLC, a workstation for programming the PLC, a SmartBits600 Ethernet load generator and Linux workstation with commonly available hackers software. As shown in Figure 1, all the equipment was connected using a standard Ethernet switch.

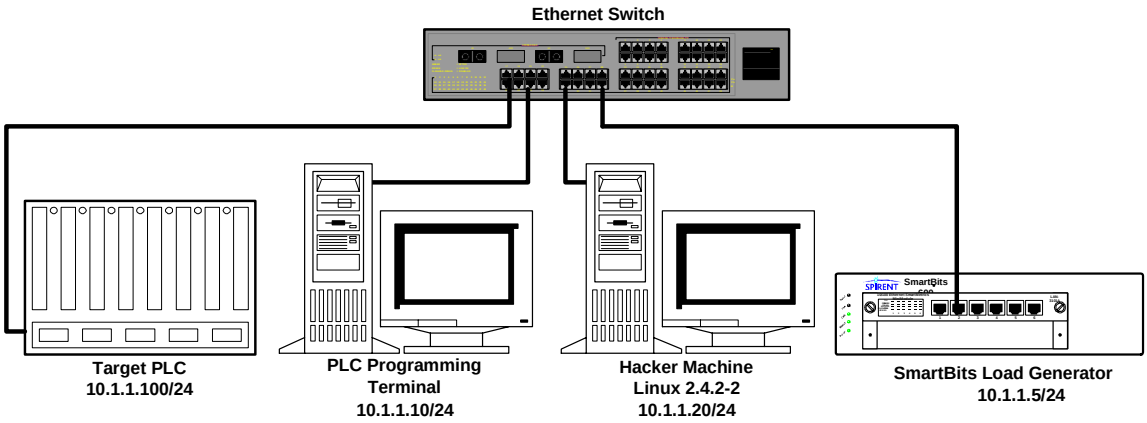

Figure 1: PLC Security Test Bench

OPEN PORTS

Internet services are identified by 16-bit integers called ports. Each TCP and UDP packet contains a source and destination port number. In packets from a client, the source port identifies the client and the destination port identifies the service. Each well-known Internet service is assigned a unique port. The protocol stack notifies a server when a packet arrives with its port in the destination port field. For example, the HTTP server is notified when a TCP packet arrives with destination port 80.

A port is "open" if a server is waiting to respond to TCP/UDP packets with that destination port. Hackers often scan for open ports, and then use an attack known to be effective on the service waiting on that port. A port scanner is an application that takes a list of IP addresses and ports and sends packets to each address/port pair, checking whether the port is open. Port scanners are readily available on the Internet and require no programming ability.

The test was performed by connecting the PLC Ethernet interface to a Linux workstation. All ports (1-65355) were scanned for UDP and TCP services using the open source utility `nmap`. For TCP, `nmap` attempts to open a TCP connection. The port is open if the connection is successfully established. For UDP, nmap sends a UDP packet and waits for an error message from the Internet Control Message Protocol. The port is considered open if an ICMP “port unreachable” response is not received.

The test results showed that a single TCP port was found open on this make of PLC. This port must be open for normal communications from programming terminal to PLC. A single UDP was also found to be open. This was port 161, a port reserved for the Simple Network Management Protocol (SNMP). An open SNMP port is potentially very dangerous, as discussed below.

SNMP

An SNMP enabled device maintains a Management Information Base (MIB) containing many fields. Each field is either read-only, e.g., TCP connection statistics, or read-write, e.g., IP address. SNMP offers a useful service, allowing a network administrator to monitor and control many network devices from a single location.

SNMP provides password protection, with one password for read-only fields and another for read-write fields. Unfortunately, the password scheme suffers from two security weaknesses (5). First, the passwords are often left unchanged from the factory defaults: typically "public" for the read-only fields and "private" for the read-write fields. The defaults are well known to the hacker community. It is easy to overlook the need to change them, especially if the installation is not using SNMP. Second, changing the default passwords is still problematic because most common version of SNMP (Version 1.0) uses no encryption. The new passwords will be transmitted over the LAN in plaintext, making them available to anyone running a packet sniffer on the network.

In this test, the PLC Ethernet interface was connected to a Linux workstation with the open source utilities `snmpwalk` and `snmpset` installed. `Snmpwalk` traverses the entire SNMP MIB, returning all fields found. `Snmpset` allows the user to change the value of any read-write field. Both utilities require knowledge of the password.

The test strategy uses `snmpwalk` to discover the MIB fields supported by the PLC. These are examined by hand to determine which fields (usually read-write) have the potential to compromise PLC operation. Then `snmpset` is used to try and impact the PLC.

The test results showed that many MIB objects were found that could render the PLC inoperable. For example, interface status (up or down) can be changed. Further, TCP/IP configuration information can be changed, effectively disconnecting the PLC from the network. In this particular PLC, no method was found to protect the PLC by disabling the SNMP services.

MALFORMED PACKETS

The purpose of this test is to check the stability of the TCP/IP stack when presented with deliberately malformed packets. It is important to note that all of these packets will have correct checksums and so will appear to be free of transmission errors.

Many of the fields in IP and TCP headers are restricted in the values permitted. Some restrictions are absolute, e.g., legal values for the four-bit IP version field are 4 and 6. In other cases the restrictions are relative, e.g., the value of the 16-bit total length field must be the same as the actual datagram length.

Malformed packets pose two risks to a PLC. First, the response to such a packet is often specific to a particular TCP/IP implementation. Thus, a hacker can send malformed packets to identify the TCP/IP

implementation in use, a first step towards compromising the implementation. Second, the implementation may fail when given a malformed packet, causing the PLC to cease functioning. For example, if the value of the total length field is 100 but the datagram length is only 80, the IP implementation may fail by attempting to read past the 80th byte.

In this test, the PLC Ethernet interface was connected to a Linux workstation with the utility `isic` installed. `Isic` is an open source utility that generates large numbers of IP packets with randomly seeded errors in IP fragmentation, version number, and header size. A series of test runs were made; in each, 6,000 packets were generated followed by a call to the `ping` utility. `Ping` checks that the PLC TCP/IP stack is minimally operational.

No errors were found from packets with fragmentation or version number errors. However, errors in header length caused serious problems. When these packets were received, the PLC exited run mode and ceased to respond to TCP/IP and serial communications. To resume operation, the PLC had to be powered down and up and its control program reloaded through the serial port.

BROADCAST STORMS

Broadcast packets are directed to all computers on a network rather than to a specific host or device. They may be generated by network servers advertising their services or by a host trying to locate a service. Broadcast messages normally use a small portion of the available bandwidth and are an important part of a properly functioning network.

In large quantities, broadcast packets can overload a network or host. The key point is that all network devices must spend some resources interpreting each broadcast packet, even if it is discarded immediately. Many devices behave abnormally if they receive too many broadcast packets in a short time (6).

Broadcast storms can be a considerable risk to PLCs. For example, several years ago an Ethernet-based PLC network in a pulp and paper mill lost communication to the operator consoles due to a broadcast storm. The cause was traced to a controller with a faulty EPROM which caused it to generate broadcast packets very rapidly. While this broadcast storm was accidental, a hacker could easily generate similar traffic.

This test is based on Address Resolution Protocol (ARP) packets. Host A sends an ARP request to get the Ethernet address of another host, say B. The request contains B's IP address and is broadcast to all hosts. Every host examines the request packet. Only the host owning the IP address (host B in this case) replies.

In each test run, ARP requests are sent at a fixed rate and the PLC behavior is monitored. The initial rate is 500 packets/second, increased by 500 for each test run. A SmartBits 6000 load generator (7) was used to generate the packets. This equipment is invaluable when a precisely metered traffic is called for. If the PLC connection is lost, then the ARP transmission is halted and an attempt made to reconnect with the PLC.

When presented with 1,500 ARP packets per second, the PLC ceased normal communications. While this is a lot of packets, on the 10 Mbps link in use, it represents only 10% utilization. This rate is easily achieved with commonly available tools and a generic PC.

RESOURCE STARVATION

Resource starvation attacks are based on normal requests for service that are issued in such large number or so quickly that the host is unable to continue normal operation. The attacks can target any communication layer: Ethernet, IP, TCP, or application. Typical TCP/IP stacks are vulnerable to a wide variety of resource starvation attacks. Usually there is a fixed limit on the number of simultaneous TCP connections. Thus, one common approach is to open so many TCP connections that normal communication is impossible.

In this test, a Linux workstation was connected to the PLC Ethernet card and normal operation was initiated. The hacker utility `jolt` was used to force closure of this connection by sending large numbers of illegal packets. Then the `netcat` utility was used to create the maximum number of TCP connections the PLC can handle, making it impossible to reopen the connection required for normal operation.

It was straightforward to force closure of the active connection. Using `jolt` for approximately 10 seconds caused this connection to time out. Then, calls to `netcat` established 255 TCP connections, the maximum supported by the PLC. At that point, it was no longer possible to establish the original connection and resume normal PLC communications.

RECOMMENDATIONS

These tests, along with the security incident database results, show that hackers have both the means and the will to disrupt DCS and PLC operations. Ten years ago that might have been unlikely since process networks were proprietary systems that were isolated from most corporate systems. Today that has changed because we are building sensor-to-boardroom integrated systems that use open standards such as Ethernet, TCP/IP and web technologies.

Depending on the corporate firewall to protect the process isn’t the answer because it ignores the fact that at least 50% of all corporate hacking is from inside the firewall. To make matters worst, there are a number reasons that standard IT security standards can’t be directly applied to the plant floor. First the nature of process control systems, with their reliance on unusual operating systems and applications, means that many of the software-based security solutions will not run, or if they do run, they will interfere with the process systems. Secondly, traditional IT security techniques focus on threats from outside the organization. As we noted earlier, this is not the primary risk for process control security. So the process control world is faced with creating its own security standards.

Where do you start if you want to build a solid cyber defense for your control system? At the present time, there are few best-practices guides or standards to guide process engineers, so the challenge is considerable. However they are not insurmountable if an organized implementation strategy is followed.

The first stage is to develop a security policy for process control systems: a statement of the goals, responsibilities and accepted behaviors required to maintain a secure process environment. The policy gives broad guidance and demonstrates senior management support for security-related facilities and actions across the organization. A security policy should be technology and architecture independent and should omit the implementing procedures and processes. In other words, the security policy outlines what you want to achieve, not how to do it.

Once the security goals are defined, an overall network architecture can be developed. This usually involves creating a multi-level network with firewalls between the layers. For example, a simple architecture might divide the plant into two levels – a business network level and a process control network level. For firewalls there a number of options to choose from. The simplest and fastest is usually a packet inspection firewall that checks each network packet against a filter list to determine if the packet should be forwarded or not. These can often be implemented directly in an Ethernet switch. More complex firewalls include proxy firewalls and air-gap systems. Regardless of which style you select, it is usually best to make it a different brand than the one used for the corporate Internet firewall.

While the firewall is the lock on the door to the process network, it is not the burglar alarm. You need some method of monitoring traffic and identifying malicious activity on the network. The tool to achieve this is known as an intrusion detection system (IDS) and can range from a simple scan detector, to a heuristic engine that profiles user behavior, to a system that takes explicit action against the suspected intruder. In the process world, traffic patterns tend to be very consistent so even simple traffic matrices that show who is talking to who can be a big help. For example, if a PC in the accounting area suddenly starts chatting up a storm to a PLC, it might be time to take a closer look. An IDS can also help you configure your firewall filters by showing what traffic patterns are normal and what patterns need to be blocked.

The layered security model is very strong if it is implemented without exceptions. Unfortunately, we all know there will be exceptions. For example, a control vendor may need to connect to a PLC via a modem to offer technical support. As tempting as it might sound, banning non-standard connections outright is not usually feasible since the primary goal is ease of production, not ease of security. What is needed is a system which can ensure that exceptions are logged and handled by means other than the standard firewall access. For example a configuration policy and tracking system of all modem connections might be a first step. A more advanced solution might be to set up a secured remote access server attached to the firewall as a common dial-in point for all vendors.

The final stage of the security strategy is to develop an incident response plan. Many times we have worked with companies that know they are being hacked but don’t know how to deal with it. Rather than waiting until they were in trouble, these firms should have established a Security Response Team and a process to deal with incidents in advance. The team would monitor events and be prepared to act quickly in the event of a serious incident.

CONCLUSIONS

Over the past ten years industrial control systems have seen a significant increase in the use of computer networks and related Internet technologies to transfer information from the plant floor to supervisory and business computer systems. At the same time, there has been an explosion in the use of Ethernet and

TCP/IP in industry for process control networks. While technologies such as Ethernet and TCP/IP allow for significant cost savings and improved interfacing for industry, it is important to understand that their origins are rooted in a culture very different from the factory floor.

Both recent industrial experience and laboratory tests of Ethernet-based PLCs clearly show the risks of adopting Internet technology without careful attention to security. With proper planning, however, the risks can be mitigated. Most important are a security policy, careful design of the network architecture, exception tracking, and an incident response team.

Internet technology has much to offer on the plant floor. The trick is to adopt the technology but not the culture.

REFERENCES

(1) KC Claffey, "Internet measurement: myths about Internet data", http://www.caida.org/outreach/presentations/Myths2002, CAIDA, UCSD.

(2) T. Stephanou, "Assessing and exploiting the internal security of an organization", SANS Institute, Mar. 2001.

(3) http://www.theregister.co.uk/content/4/22579.html.

(4) E.J. Byres, "Network secures process control", InTech, Instrument Society of America, pp. 92-93, Oct. 1998.

(5) L.D. Paulson, "Protocol-related problem threatens Internet security", *Computer*, pp 20, Apr. 2002.

(6) Cisco Systems Inc., *Internetwork Design Guide - Appendix E*, 1997.

(7) Spirent Communications, *Smartbits Applications Notes*, 2002
http://www.spirentcom.com/analysis/productView.cfm?D=5&T=37&WS=7&P=114