Tools for Cost-Effective Authentication

B.D. Geelhood, R.R. Hansen, R.T. Kouzes, W.K. Pitts
Pacific Northwest National Laboratory
PO Box 999, MS P8-08, Richland, WA 99352, USA (509) 376-0276 bruce.geelhood@pnl.gov
June 8, 2001

ABSTRACT

This paper examines the requirements and motivations for development of authentication tools for monitoring systems. *Authentication* is the process by which the Monitoring Party gains appropriate confidence that the information reported by a monitoring system accurately reflects the true state of the monitored item. Gaining confidence that the inspection results are always credible is a greater challenge when an information-barrier, which protects classified information by blocking access to raw data and intermediate results, is included in an automated measurement system. Confidence in the results requires gaining continuity of knowledge regarding all the data processing within the entire system. The authentication process involves searching for both 1) inadvertent design or implementation flaws leading to incorrect results or a non-robust system and 2) deliberate covert features designed into the system for some Host advantage.

Functional testing can only address the first of these. A thorough investigation of system functionality can ensure it functions as designed over a wide range of measurement parameters and environmental conditions. It is important that the analysis software be robust enough to handle unexpected situations, since these systems are designed to operate in a closed and secure mode without user input or analysis. Functional testing tools include a reasonable set of test sources and a means of combining data from them to span a wider range of input parameters. Software that distorts spectra in a controlled fashion can establish robustness limits.

Thorough authentication must address the threat where the host selectively triggers a built-in "hidden switch" that alters the measurement results to erroneously pass selected containers. Functional testing is insufficient to preclude a selectively triggered "hidden switch" unless the host is compelled to trigger it during that testing. Private examination of a duplicate system with complete documentation and system design information is crucial to discovering intentional flaws. The host can be deterred from installing a "hidden switch" by both the documentation requirements and a random selection of duplicate components for private examination. *The Host's ignorance of the Monitor's activity during private examination is also a powerful deterrent to cheating, since the Host cannot be certain that a hidden switch will remain undetected*. Matching the actual system to the duplicate and documentation is an effective tool for finding hidden switches.

INTRODUCTION

An automated measurement system can be used to make a non-intrusive measurement during a verification inspection to confirm compliance. When the measurement potentially divulges information considered classified by the Host, an information barrier can be used to protect that information. The basic, top-level functional requirements [1] for the information barrier portion of an integrated radiation-signature, information-barrier-protected inspection system are twofold:

1. The Host must be assured that his classified information is protected from disclosure to the Monitoring Party, and
2. The Monitoring Party must be confident that the integrated inspection system measures, processes, and presents the radiation-signature based measurement conclusion in an accurate and reproducible manner.

These requirements can both be achieved under a host-supply scheme, where the Host has last private access to the system prior to use, as a means of providing the classified information paramount protection. When the Host designs and supplies a transparent measurement system, the Host gains confidence that the Monitoring Party has not included some covert feature to divulge classified information. This aids the Host's certification team, which is tasked with ensuring that sensitive and classified information is protected throughout the measurement process. In exchange, the Host's design must aid the Monitor's authentication team, which is tasked with gaining confidence that the inspection results are always credible. The information-barrier, which protects classified information by blocking access to raw data and intermediate values, is best implemented through use of a transparent system, where all the details of the data processing within the system are available to all parties. Without an information barrier, confidence in the results is often obtained by repeatedly observing consistency between 1) theoretical expectations, 2) raw data, 3) intermediate values and 4) the final numeric result. The information barrier protects classified information by blocking access to all potentially classified information by presenting only pass/fail results regarding pre-agreed criteria. Without that intrusive access, confidence in pass/fail results comes from continuity of knowledge regarding each data processing step throughout the entire system. Although some of the data used are classified, the data processing itself is unclassified and can be shared.

Authentication is the process by which the Monitoring Party gains appropriate confidence that the information reported by a monitoring system accurately reflects the true state of the monitored item. Thus, the authentication process involves searching for both 1) inadvertent design or implementation flaws leading to incorrect results or a non-robust system and 2) deliberate covert features designed into the system for some Host advantage.

The potential for a deliberate covert feature is a threat to credibility. A covert feature in a system can be analogous to a hidden microphone in a conference room. Since the technology exists for covert features, prudence requires both a search for and an attempt to mitigate potential problems without accusation. The Monitoring Party is generically concerned with the potential for a selectively triggered hidden switch, which the Host can use at will to covertly switch failing results into passing results. Another concern is a denial of services problem where the system has either a low operational reliability or a high false negative rate. Both make it very difficult to obtain the desired statistical confidence. The potential sophistication of hidden features that could be introduced by the Host should not be underestimated. If the hidden switch could not be selectively triggered, functional testing might readily discover it. However, if the Host merely fails to trigger the switch during functional testing, the risk of discovery can be negligible. A prudent authentication team will search the entire system for 1) a receiver for a trigger signal, 2) the pathway to the switch, and 3) the switch. These components can be hidden anywhere in hardware or software. The trigger may not require a separate receiver or pathway. Subverting the software is especially easy if the Monitoring Party lacks continuity of knowledge regarding each

data processing step since results can be easily altered by the insertion of only a few instructions. For example, a hidden switch can easily determine if the measured plutonium mass is close to some specific low-mass value to trigger replacement of the actual failing value with a higher-mass passing mass value.

This paper is a partial listing of authentication tools that can be used to make the authentication process more efficient. This paper acts as a brief summary of technical documents produced by Pacific Northwest National Laboratory (PNNL) on design guidance to facilitate authentication [2], authentication tools [3], and authentication procedures [4]. The degree of credibility obtained in a system because of authentication activities is a function of cost. Credibility will be low until functional testing and a thorough understanding of the system is achieved. These activities require some unavoidable baseline funding level. Then credibility increases with funding as a thorough search for potential hidden switches is conducted. However, costs can increase exponentially when pressing for the last portion of absolute credibility. Tools that automate or facilitate functional testing and/or careful examination can be very cost effective.

DESIGNING FOR AUTHENTICATION

A mutually agreed upon open or transparent design can greatly aid the Monitoring Party's authentication efforts. In general, a design that facilitates easy access for visual inspection and eliminates extraneous functionality is most easily authenticated. Incorporating tamper-indicating devices (TIDs) and enclosures (TIEs) can facilitate re-authentication efforts, because they could preclude unauthorized modifications to the system between inspection visits and usage and provide continuity of knowledge. Cooperation between the design team and the authentication team prior to and through the design review is very beneficial. When the cost of designing, building, and authenticating a system is viewed as a whole, it is easier to appreciate that design decisions should include balancing authentication resources with design resources. For example, use of a software module without source code that the design team is familiar with may slightly reduce the design cost, but force the authentication team into a very costly reverse-engineering effort. Thus, a priority should be placed on the availability and completeness of documentation when selecting components during the design phase.

PNNL has written an extensive guidance document regarding the design of systems to facilitate authentication [2]. Examples of illustrative design features aiding authentication include:

- Avoiding massive software operating systems and I/O libraries reduces the volume of code subject to time-consuming examination.
- Implementing software to execute in place in a once programmable read-only memory (PROM) precludes self-modifying software and unexpected overwriting of executing software. Using a PROM rather than magnetic media precludes overwriting software files.
- Disabling unused I/O and interrupt lines by hardwiring to ground precludes unauthorized or unexpected use to trigger a hidden switch.
- Implementing a hash function comparison of the installed software precludes modification of the installed software by providing an *in situ* method of matching it to the baseline.
- Avoiding the use of multiple and different computers could reduce examination requirements to a single operating system and BIOS software version.
- Using printed circuits rather than wire-wrap or hand-wired circuits reduces the number of continuity measurements and facilitates photographic comparisons.

- Using a spacious mechanical design that facilitates disassembly and re-assembly could aid physical inspections.
- The system design must support an open mode in a manner that prevents the system from sensing if it is in open or secure mode.

DETAILED DOCUMENTATION

Detailed and thorough documentation is crucial to the authentication effort as a means of achieving continuity of knowledge regarding the data processing performed by the automated measurement system. A design priority should be placed on providing the level of detail necessary to discover both inadvertent flaws and intentional covert features. Promptness and change controls are important under a tight schedule.

High-level descriptions of various hardware and software modules, and various design documents are very useful in providing instructions and conceptual explanations regarding the functionality in each module. However, if a covert feature is deliberately inserted into a module, the overview documentation undoubtedly will not mention it. In the search for covert features, the most useful documentation is that which is forced to include information on all features, both overt and covert.

For software, the commented source code is the highest level of software documentation that is forced to include all covert features. The completeness and accuracy of the source code can be independently verified when the provided source code is built into executable code, which exactly matches the installed executables. The independent build of the executable software is necessary before expending resources on detailed examination of the source code searching for covert features. Thus, the software documentation package for each module should include at least:

- Commented source code for all software residing on the automated measurement system
- Build instructions for producing the baseline executables from the provided source code
- Copies of all the compilers and other material required for building a duplicate executable
- A baseline copy of all installed executables for comparison to independently built executables and actually installed executables
- Explanations of the data collection (commands & formats) and analysis algorithms

Without provided source code, the authentication team could in principle reverse-engineer the software from executables extracted from the duplicate system provided for private examination. While software tools are available for reverse engineering, this process is expensive and slow. It is much more desirable and cost effective for the design team to consider the availability of source code as a primary selection criterion.

Some commercial software may not require detailed examination of the entire source code, if it was written prior to the design of the system. It may be adequate to show such installed executable code identical to code independently and anonymously procured in a mass market. However, any commercial software may have undocumented features that could be used to subvert results. Any extraneous functionality could be overwritten for other purposes. The I/O software that passes information into and out of the system is of particular concern as a potential trigger pathway and should be examined in detail.

For hardware, the analogous documentation package forced to include covert features is more diverse because the hardware itself is more diverse. The analogous hardware documentation package for each module includes at least:

- Detailed photographs of all the components at board level
- Complete schematics and wiring diagrams
- Complete parts lists with vendor information included
- Construction details (e.g., circuit-board layouts)
- Programming for all custom components and the means to independently verify the design

Likewise, the completeness and accuracy of the hardware package can be verified by detailed comparison to the actual or duplicate hardware. Software to independently build duplicate programming information for all programmable components from source files is an essential authentication tool. It is also crucial that the design does not implement security measures precluding the readout of logic-device programming.

JOINT INSPECTION TOOLS

The period of joint inspection prior to the start of any measurement campaign is necessary to establish that the system 1) remains unmodified and 2) is properly set up. A negotiated protocol will specify the level of equipment access, and limit the duration, the number of personnel, and functions that can be performed. Authentication tools can and should be utilized to make this process as efficient as possible. The Host has valid security concerns that the Monitoring Party must consider. Tools exercised during joint inspection include:

- **TAMPER-INDICATING DEVICES** – TIDs and TIEs can indicate if the system has had an unauthorized access; but they do not reduce the requirement for a thorough initial inspection. Inspection and reapplication of TIDs would be a part of any joint inspection protocol. The ability to photograph TIDs and TIEs and remove used TIDs for private examination enhances the value of these tools. However, TIDs and TIEs are vulnerable and do not enjoy absolute trust. The degree of confidence that the system remains in a baseline state, matching the documentation and an initial duplicate, can be enhanced with the following authentication tools.
- **RANDOM SELECTION** – Random selection is the single most powerful and useful authentication tool. It can be used during inspection visits to provide the Monitoring Party with exact duplicate components for private examination and also to mitigate Host concerns that an examination technique might alter the system prior to use. For example, a duplicate software-bearing component facilitates a byte-for-byte comparison with the Monitor's copy of the baseline software and removes any possibility of altering the installed software. However, random selection consumes components more rapidly than failure replacement. To facilitate random selection, it is essential that all software-bearing components be socket mounted rather than soldered in place. In an example random selection scheme, the Host provides multiple identical copies of a component from secure and joint storage (i.e., spare parts locker). The Monitor ensures that these are all uniquely marked to preclude sleight-of-hand swaps. The Monitor selects one for private examination and another for the Host to install under observation. Any remaining components are returned to storage. The Monitor's choice is considered random because the Host has no prior knowledge of a components future. It could be placed into the system or removed from the facility and subjected to intense

scrutiny. A photograph of the marked and installed component could be used to confirm that it remains in place until swapped out in the future. The Monitor's component could be examined on-site or promptly removed via diplomatic courier for private off-site examination, which should deter the Host from using non-standard components or altering the software. All the software-bearing components could be initially installed using random selection and subsequent inspection visits could focus on one Monitor-selected subsystem or component for replacement using random selection. If an entire subsystem is replaced, additional confidence regarding the hardware components is gained. A Russian paper [5] suggested a variation that allows private examination of a replaced component. This recognizes that classified data can never reside in a previously written non-erasable programmable read-only memory, especially if the system has no provision for writing to it. If a TID protected the replaced component, previously used software could be examined. Examination of the replaced component is best used when focused on one subsystem without prior notice, because all the remaining software is forced to match baseline by the *de facto* offer for private examination. The most confidence is achieved when the Monitor selects a single software-bearing component as the focus, selects a spare for private examination, and receives the replaced component.

- **HASH FUNCTION** – A scheme for verifying that the system software remains unmodified based on an *in-situ* secure hash function comparison can be acceptable, but implementation details are very important. Hash function schemes are often not simply explained or easily understood. In simple terms, a Hash function produces a unique digital signature for computer code, with the unique signature depending upon both the input seed and the computer code. Host supply of the hash function programming should mitigate reasonable security concerns regarding numeric input and output from a computer previously used for classified processing. However, evaluation of hash algorithms and implementations are a prerequisite for acceptance.
- **PHOTOGRAPHY** – Photography of hardware components down to board level components during joint inspection allows the authentication team to very efficiently examine the hardware for unauthorized modifications. Comparing current photographs to a library of photographs from 1) provided documentation, 2) private examination of duplicate systems, and 3) prior visits is a very important authentication tool to preclude unauthorized modification. The photographic resolution should be high enough to spot potential problems. A high-resolution digital camera or a high-quality self-developing film camera will allow the Monitoring Party to ensure the quality of each picture and the Host to mitigate security concerns.
- **X-RAYS –** X-rays of board level components can provide details regarding internal traces on multi-layer printed circuit boards and could discover any non-standard and/or undocumented components. An X-ray is an effective substitute for visual inspection of components within sealed containers (e.g., the HPGe sensor in the vacuum Dewar). An X-ray could be an effective substitute for private examination of certain high value items such as a HPGe sensor.
- **VISUAL INSPECTION –** Visual inspection of the hardware by the Monitoring Party's team prior to each measurement campaign allows discovery of obvious problems. This inspection is most useful if comparisons to previous photographs of the duplicate system are made during the initial inspection or private examination. Visual inspection with a checklist is sufficient to ensure that the system is properly set up for the measurements (e.g., reasonable parameter values, proper jumper settings, and proper interconnections). Visual inspection is not a substitute for photographic comparisons to discover subtle changes.

- **SELECTED ELECTRICAL MEASUREMENTS** – A joint inspection team could make electrical measurements on the operational system to verify correct operations. Test points could be examined with a battery-operated portable oscilloscope for correct voltage levels and waveforms. This joint measurement process may be time limited (e.g., one hour). The provided documentation would be used to pre-select the test points and measurements to confirm correct setup and operation (e.g., time constants, pole-zero corrections, and signal shapes). These measurements might include insertion of test pulses from electronic sources to evaluate the system and independently confirm its calibration.

JOINT FUNCTIONAL TESTING TOOLS

The period of joint functional testing prior to the start of any measurement campaign is necessary to establish that the system 1) functions as designed (i.e., remains undamaged) and 2) is properly calibrated. A negotiated protocol will specify the level of equipment access and limit the duration, the number of personnel, and the functions that can be performed. Functional testing is insufficient to preclude a selectively triggered "hidden switch" unless the Host is compelled to trigger it during such testing. The limited time and huge number of input combinations and sequences result in a negligible probability of discovering an intentional and sophisticated selectively triggered "hidden switch" by exploratory functional testing. Functional testing tools include:

- **OPEN MODE TESTING** – The ability to operate the system in an open mode with the ability to view numeric values from measurements of calibration sources is a crucial tool in determining that the system remains undamaged since last use. A secure mode measurement with only a pass/fail output is unlikely to reveal a progressive system degradation problem. For example, loss of gamma-ray resolution associated with neutron damage over time affects system credibility. Access to raw data, intermediate values, and numeric results can enhance credibility that the system 1) functions as designed and 2) is properly calibrated. The system design must support open mode in a manner that prevents the system from sensing if it is in open or secure mode. Without open mode, many authentication tools are rendered useless.
- **CALIBRATION SOURCES** – The Host supplies a set of unclassified radioactive calibration sources used to functionally test and calibrate the system. Setup validation includes energy calibration for the gamma-ray spectral analysis and requires measurement control for a neutron multiplicity counter. These radiation sources can also be used to confirm that the system passes and fails appropriate sources correctly for specific attributes, since the set will contain items designed to both pass and fail attributes. The ability to independently measure these calibration sources on a system with access to the raw data and numeric values increases confidence that the sources are as declared.
- **CHALLENGE MEASUREMENTS** – To prevent an obvious covert feature, which passes all assays in secure mode, the Monitoring Party reserves the right to occasionally and without notice insert a secure-mode measurement of a Monitor-selected calibration source into the inspection agenda. In addition, this random disruption of the measurement sequence increases the risk associated with a scheme that passes items based on the sequence of events.

PRIVATE EXAMINATION TOOLS

Private examination in a Monitor's facility of the duplicate system and various subsystems procured independently or by random selection from host-supplied duplicates is a very powerful authentication tool. This removes resource and protocol limitations experienced on site. Private

examination allows 1) searching for a selectively triggered hidden switch, 2) determining the limits and robustness of the analysis, 3) expanded inspections with electrical measurements, and 4) exploratory testing. The uncertainty regarding authentication activities and capabilities during private examination can be a deterrent to potential Host cheating. These brief and general tool descriptions show the serious consideration given to authentication and are only illustrative examples of useful tools.

- **INDEPENDENT PROCUREMENT** – Prior to receipt of a duplicate system, the authentication team can independently procure likely and declared components of the system for private examination. Showing that the installed components exactly match independently procured components is valuable for hardware and software components that can be anonymously purchased in a mass market. Early examination of components 1) provides a basis for specification and design reviews, 2) precludes acceptance of non-standard components, and 3) provides knowledge to plan and implement robustness tests, covert feature searches, and on-site inspections. Purchase of vendor's manuals, documentation, and development kits also facilitates authentication efforts.
- **SOURCE CODE EXAMINATION** – Automated tools are available to aid the detailed examination of software source code for covert features. Comparison programs, search features of editors, legacy code analyzers, code disassemblers, emulators, and logic analyzers can all be used to advantage during private examination to search for problems.
- **ROBUSTNESS TESTING** – Analysis software can be stressed by application to distorted data. Software that manufactures sets of modified experimental data (e.g., altered energy calibration, degraded peak resolution, and altered peak shapes) facilitates this process. Especially useful is software that combines data from a limited set of test sources to create an expanded set of test data that more completely spans a wider range of experimental parameters.
- **RANDOM SELECTION** – Random selection, described previously, is a means of obtaining an initial duplicate system and various duplicate subsystems over the lifetime of the regime.
- **HARDWARE EXAMINATION** – A baseline set of photographs and X-rays can be developed for automated and manual comparison to those obtained during joint inspections. A baseline set of electrical and electronic measurements can be developed for use in understanding the system and for planning subsequent joint inspections. Layout and design software, emulators, and logic analyzers are all normal engineering tools used to understand hardware operations and find problems. Many of these may not be exportable and can only be used during private examination. Tools are being developed to read out and compare the programming of custom logic and other programmable devices.
- **TARGETED FUNCTIONAL TESTING** – Several features of the design are included to preclude covert features (e.g., power line filtering). These features can be subjected to testing to determine their effectiveness. Additional measurements can reveal and explore potential covert signal paths through the system. Any potential trigger pathways or covert features identified by software examination can be explored and demonstrated. Potential spoofing and interfering radiation sources can be measured, and the results examined. A valuable tool is the ability to synthesize potential problem data sets rather than fabricate actual sources.

SUMMARY

Authentication primarily addresses the credibility of the results from an automated measurement system. The authentication team must develop and use several tools to gain confidence in the results. The task can be facilitated by cooperation with the Host's design team, to both design-in

authentication features and provide a quality documentation package. Credibility can also be enhanced with protocols that allow adequate system access and the use of meaningful authentication tools. Authentication is achievable with adequate system access and cooperation.

ACKNOWLEDGEMENTS

This work was supported by the U.S. Defense Threat Reduction Agency and by the U.S. Department of Energy. Battelle operates Pacific Northwest National Laboratory for the U.S. Department of Energy under contract DE-AC06-76RLO 1830.

REFERENCES

[1] Joint DoD/DOE Information Barrier Working Group, *Functional Requirements for Information Barriers*, PNNL-13285, Pacific Northwest National Laboratory, Richland, WA, May 1999.

[2] B.D. Geelhood, R.R. Hansen, R.T. Kouzes, W.K. Pitts, et al., *Guidelines for Designing Systems that Can Be Authenticated*, PNNL-13386, Pacific Northwest National Laboratory, Richland, WA, November 2000, updated June 2001.

[3] R.R. Hansen, B.D. Geelhood, M.A. Knopf, R.T. Kouzes, et al., *Authentication Methods and Tools*, PNNL-13558, Pacific Northwest National Laboratory, Richland, WA, June 2001, revision 0.

[4] R.A Warner, D.E. Robertson, et al., *Procedures for Authentication of Mayak FMSF Monitoring Equipment*, PNNL-13549, Pacific Northwest National Laboratory, Richland, WA, June 2001, revision 0.

[5] N. Rubanenko, *Report on Project Development of Design Principles and Strategy for Using Information Barriers as Applied to Radiation Technologies for Controlling Weapon-Grade Nuclear Materials*, RFNC-VNIITF – PNNL Lab-to-Lab Task 2.2 Deliverable 3.2, 2001.