Consistency Under Deception Implies Integrity

From Fred Cohen & Associates - Analyst Report and Newsletter

Consistency analysis has been found useful in detecting corruptions of all sorts, ranging from accidental bit flips (i.e., parity checking) in the 1960s, to multiple bit error detection (i.e., cyclical redundancy checks), to malicious alteration detection (i.e., cryptographic checksums)[1] in the 1980s, and has been used in digital forensics since at least the 1990s[2]. All of these are in the digital space. But as the digital and analog spaces increasingly converge in industrial control systems, complex corruptions of the combined digital and analog spaces are being used to induce harmful physical effects through exploitation of the combined systems. As an approach to defeating limited attempts to alter control system, effector, and/or sensor signals, the notion of consistency checking again comes to the fore.

Computational leverage

Notionally, the use of cryptographic checksums for detection of intentional alteration gains its utility from the computational leverage of detection over forgery. The forger not holding the cryptographic key to generate a true cryptographic checksum for any desired bit sequence is unable to systematically forge sequences in which the cryptographic checksum is consistent with the content it covers. While replay attacks and similar methods may function in systems not well designed to defeat them, such systems can be and have been designed and are successful in mitigating attacks, up to the point where the attacker is able to determine the cryptographic key, at which point forgery becomes feasible and inexpensive. The selection of the cryptographic checksum method is intended to be such that this takes a long time, and thus without infeasible computational capacity, the attacker cannot systematically forge for a long enough time to make the system secure for the intended use. Or at least this is how it can be designed.

Notionally, consistency checking in digital forensics also gains its utility from computational advantage. In this case, normal operation of computer systems produces redundant traces, and these traces can be compared for consistency. While trivial forgeries can function against detectives who are unfamiliar with the consistency methods available today, the complexity of creating a forgery that cannot be detected in the larger overall system is thought to be so high that it is infeasible in almost all cases.[3] Again, the forger cannot anticipate and alter enough traces to defeat all feasible consistency checks, and the alteration of these traces introduces potential inconsistencies with still other traces that are also subject to detection.

Thus, consistency checking leverages computational advantages of defenders over attackers. In doing so, it refutes the common but false assumption that the defender has to protect against all possible attacks in detail while the attacker only has to find one attack the defender failed to defend against. Note also that caught attackers don't always get to keep trying.

[1] F. Cohen, "A Cryptographic Checksum for Integrity Protection", IFIP-TC11 "Computers and Security", V6#6 (Dec. 1987), pp 505-810.

[2] F. Cohen, “A Note on Detecting Tampering with Audit Trails”", 1995

[3] F. Cohen, “Digital Forensic Evidence Examination”, ASP Press, 2009-2011

Detection Under Normal and Altered Operating Conditions

Under normal operating conditions, a failure in a sensor, effector, or other system component will be reflected in altered signals indicative of a move away from the weighted “center” of the control envelope (i.e., the set point). The response from the automated control system will be to compensate by altering effectors so as to move the system back toward the center of the control envelope. Of course there may be some losses (e.g., leaked fluids) resulting from the actual fault (e.g., a hole in a pipe), and the control system will continue to compensate as well as it can, and potentially alert operators to the fault condition (e.g., fluids are running low).

If an attacker alters a sensor to produce false information, the response from the automated control system will be to compensate by altering the effectors in response under the assumption that the information is true. Thus, a reflexive control attack is realized as the reflexes of the control system react to the information available. To some tolerance, the changes will be within the control envelope of the system and stability will be retained, even if some other performance effects may occur (e.g., undetected theft of fluids or excess line voltage). As tolerances are exceeded, the control envelope may be exceeded and the system may become unstable, may collapse, and may, as a side effect, destroy physical components.

However, if there are multiple sensors, to the extent that unaltered sensors are effected by control changes based on false information, the unaltered sensors may generate signals inconsistent with the altered sensor signals, supporting inconsistency detection. If redundant sensors are separate and different, common mode failures may also be avoided and better diagnosis supported. The same is true of an altered effector or an altered physical system, and to the extent that there are redundant control systems, to an altered control system. Thus, the potential exists to use the control system to detect inconsistencies between sensors, perhaps diagnose the most likely bad sensor(s) and overall situation, and in an advanced system, perhaps compensate for and reduce the trust and dependence on the false signals.

At some level of induction and/or suppression of signals (i.e., alteration), so many signals may be altered that an entirely false picture that is itself consistent may result. If every system and component is taken over by an attacker, the system may not detect or report anything. But even short of this, the deception may be of sufficient quality as to mimic the legitimate control system and deceive the operator and the systems they depend upon.

Induction and suppression of signals (i.e., deception) for detection

To compensate for this class of attack, an alternative approach is to intentionally induce and suppress the normal control signals that would be used in a systematic way so as to produce systematic changes in the overall system that (1) remain within the safety margins of the control envelope and (2) produce time variant effects within and throughout the system under control. By doing so, the sophisticated control system may induce changes that ripple through the system as a form of diagnostic test, creating sequences of alterations that remain safe and relatively efficient while inducing feedback that reveals attempts to circumvent normal controls. By doing so in an externally unpredictable sequence, the malicious actor wishing to alter the control system may be detected if they are unable to predict the proper control signals in time to reflect a globally consistent system state and variance, even though they have control of most of the sensors and effectors. Thus, computational advantage is used by the control system designer to detect and potentially diagnose malicious alteration.

Limitations of these methods

There are two cases to consider; a quasi-static case in which the system is assumed to be over damped and fed by more or less constant volumes relative to measurement time frames, and a dynamic case in which the system is under-damped and constantly changing, so that waves normally build upon each other.

The quasi-static case:

In the quasi-static case, subject to small-time delays associated with the propagation of change, conservation of mass dictates that the volume in a repository is equal to the sum of all the flows into that repository. Because the system is quasi-static, the conservation rule can be applied to within the measurement precision of sensors, and variations detected as inconsistent if they exceed the variances in precision, again subject to the relatively small propagation delay. The detection time for a leak is dictated by the precision of measurements, so that a device that measures a water tank to the nearest 1000 liters will detect leaks totaling no more than 2000 liters essentially as it happens.

For a system of tanks and pipes, the height of each stage in the system should match the flows in and out, and an attempt to provide false signals for any single sensor will result in a detection as soon as the leaks create a mismatch. By altering multiple sensors, it is feasible to shift apparent usage from one area to another only to the extent that it doesn't create a downstream inconsistency. Thus, at downstream endpoints, a malicious actor could steal water and place the blame on another party, but theft in the middle of a system would be detected by the downstream loss, and in order to avoid such detection forged signals from downstream sensors would be required in the proper combinations so as to create a consistent resulting overall system.

Time to detection depends on the volumes flowing and sensitivity, so that for 2000 liters of loss in a system flowing 200 liters per second, 10 seconds plus propagation time plus sensor scan time limits the time till detection. However, noise factors such as rain and wind may effect volumes and sensor precision. For example, if it rains, the system will gain water that could be stolen as it is gained.

For passive detection, consistency analysis allows forgeries by parties who understand and can model the system reasonably well, perhaps including taking sensor readings from other (downstream) sensors and forging multiple values so as to retain consistency of the overall system in near-real-time. By adding active alterations to the system through actuator changes, the system can further resist modeling by malicious actors. For example, instead of having a fully predictable control system that seeks to keep water in tanks at constant set points, suppose the system intentionally changed set points over time. Now the attacker seeking to model the system has to take into account the changes in set points in order to create a consistent forgery. Instead of simply forging sensor readings to cover up a loss for a time, the attacker has to calculate the proper values for the entire downstream system taking the changes in all set points into account, or the forged values will be internally consistent, but inconsistent with the changes set points. The control system's model of the overall system is unchanged, and detection remains the same problem, while forgery becomes far harder and requires access to and analysis of all of the changes set points for effect.

The dynamic case:

Unlike the quasi-static case, an underdamped system with changes that don’t have time to propagate to stability produces a far more complex challenge for both attacker and defender. Conservation remains true, of course, but measurement becomes far harder. A sensor measuring a wave form may vary significantly from the average value, and integration does not produce a reliable mean value in an underdamped system over a short time frame. Creating additional dynamics is far more problematic in that, without complete knowledge of the system state over time, induction of changes may force the system out of stability and generate positive feedback, ultimately resulting in catastrophic failure. Sensors may be at nodes in the system providing incorrect feedback, and perhaps more importantly, the utility gained in detection may be nullified by the large variance in sensor values in different operating modes.

On the other hand, to the extent that these problems can be solved, the situation for the attacker is potentially far more complex. Now they have to launch their attack in an environment where their changes may produce system instabilities that make their changes obvious quickly, they have to compute values upstream and downstream, and the computation of effects of alterations may take too long for real-time analysis. The defender has the advantage of proposing a change, calculating expected values with time through a complex analytical process, and then using actuators in combinations and sequences so as to produce predictable dynamics. The attacker without advanced knowledge of the planned changes is forced to try to do real-time analysis of the effects of the changes and produce solutions in time to forge sensor data to within the fidelity required to fool the defender's predictive system.

Limits based on relative volumes:

A further limit is worth pointing out. Intentional variations in flows for detection may be problematic in systems such as power infrastructure, where back forces and phase shifts may result from attempts to alter flows. While at distribution points, such changes are relatively straight forward with smart meter technologies and force levels are relatively low, if many such changes are made in concert, the combined forces may be enormous. Thus, scheduling of changes may become a serious challenge for the overall system. In transmission, shifting enough power to produce changes in excess of detection thresholds may require so much energy that the system becomes less stable, back forces on generation may be problematic and damaging, interactions of wave forms may cause nodes in the system in excess of allowable tolerances, and compensation for rapid changes in load that occur all the time may force the detection thresholds to be set so high as to make large-scale dynamic detection infeasible without risking system stability. Computation of changes and detectable effects may be so complex as to make dynamics infeasible.

Conclusions:

Consistency analysis appears to be a feasible method for detecting intentional acts altering control systems, and intentional deception in the form of induction and suppression of signals can be used to gain computational leverage over attackers. In the quasi-static case this is now feasible, while in the dynamic case it is far more complex and potentially dangerous, but also potentially far more advantageous to the defender.