atsec the information security provider

NEWS 02/14

Wer einmal lügt ...

atsec verhilft seinen Kunden zu mehr Informationssicherheit. Die Grundlage unseres Geschäfts ist dabei das Vertrauen, das unsere Kunden in unsere Arbeit und unsere Integrität haben. Natürlich müssen die Ergebnisse, die wir abliefern, nachvollziehbar sein – nicht nur für unsere Kunden, sondern oft auch für deren Kunden und darüber hinaus. Nachdem wir aber oft dort ins Boot geholt werden, wo wir eine spezifische Expertise einbringen, die der Kunde nicht hat oder wo es viel zu aufwändig für ihn wäre, unsere Erfahrungen und Einschätzungen detailliert nachzuvollziehen, läuft es doch letztlich darauf hinaus, dass wir für vertrauenswürdig genug gehalten werden, diese Arbeit zu tun.

Was heißt aber „vertrauenswürdig"? Der Duden nennt als Synonyme u. a. aufrichtig, ehrlich, glaubwürdig und rechtschaffen. Das sind Begriffe, die wir mit unserem Unternehmensziel der Integrität verbinden und die wir für unsere Arbeit auch so in Anspruch nehmen. Wie aber können wir dies vermitteln? Ein tiefer Blick in die Augen reicht da nicht – jedenfalls nicht immer ☺. Externe Audits und unsere Zertifizierungen nach ISO 9001, 27001 und 17025 zeigen unseren Kunden, dass wir nach dem Stand der Technik arbeiten. Vertrauen ist jedoch mehr: Wir müssen in allen unseren Handlungen immer integer sein. Ein einziger Fehltritt bedeutet hier unweigerlich den Absturz.

Schwierig wird es, wenn unsere Arbeit nur ein Teil einer größeren Vertrauenskette ist, die wir nicht vollständig kontrollieren. Dies ist z. B. bei Evaluierungen nach Common Criteria (CC) der Fall. Das CC-Zertifikat, das uns Vertrauen in die Sicherheit eines IT-Produkts vermitteln soll, basiert auf einer ganzen Vertrauenskette: Produkt > Hersteller > Prüfstelle > Behörde > Zertifikat.

Obwohl „unsere" Vertrauenskette über unsere europäischen Prüflabore und Zertifizierer nach wie vor intakt ist, haben wir es derzeit mit einer massiven Vertrauenskrise zu tun, weil dies in anderen Nationen, mit denen Abkommen zur gegenseitigen Zertifikatsanerkennung bestehen, nicht mehr der Fall ist.

Wir sehen, dass z. B. in den USA die Zertifikate von NIAP/NSA ihre Vertrauenswürdigkeit verloren haben, weil die Kette bei der NSA gebrochen ist. „Vertraut uns – wir sind die Guten" funktioniert nicht, wenn man kein Partner, sondern ein Ziel ist. Das betrifft derzeit besonders die Five Eyes, prinzipiell aber alle Nationen, in denen CC-Zertifikate von der „intelligence community" vergeben werden, die sich nicht als „intelligent community" präsentiert, weil sie nicht gemerkt hat, dass die eigene Vertrauenswürdigkeit von der Integrität des eigenen Handelns abhängt. BSI-Vizepräsident Könen hat es neulich gut auf den Punkt gebracht: Es muss klar sein, wer Offense und wer Defense spielt. Das BSI positioniert sich dabei klar in der Defense. Bei anderen ist das höchst unklar. Ist ein Zertifikat einer Nation vielleicht nur ein übler Trick, um uns Hintertürchen für Schlapphüte unterzujubeln? Ohne Vertrauen hat die gegenseitige Anerkennung der CC-Zertifikate auf internationaler Ebene vorerst ausgespielt. Jetzt muss der Markt entscheiden, wessen Zertifikaten er noch vertraut. Damit das nicht jeder für sich in jedem Einzelfall tun muss, hoffe ich, dass sich jenseits des CCRA bald eine neue CC-Liga formiert, in der die Vertrauenskette aufrecht, ehrlich und rechtschaffen zusammengehalten wird.

Herzlichst,

Gerald Krummeck
Leiter Prüfstelle

Veranstaltungen und Messen 2014

Die atsec information security wird in diesem Jahr auf folgenden Messen und Veranstaltungen vertreten sein:

- **it-sa in Nürnberg: 07. bis 09.10.2014**
  Auf der größten IT-Sicherheitsmesse im deutschsprachigen Raum wird sich atsec auch im Jahr 2014 wieder mit einem eigenen Stand präsentieren.
  Sie finden uns in Halle 12.0 am Stand 214.
  Wir freuen uns auf Ihren Besuch.

- **Workshop ISO/IEC 27001 in München: 29.10.2014**

  atsec führt für die Allianz für Cybersicherheit einen kostenlosen Workshop zum Thema „Aufbau eines ISMS auf Basis von ISO/IEC 27001" durch. Auch unsere Bestandskunden sind dazu herzlich eingeladen.
  Weitere Details dazu finden Sie auf Seite 4 dieses Newsletters.

atsec it security blog
Verfolgen Sie brandaktuell Diskussionen und Erlebnisse unserer Mitarbeiter auf
http://atsec-information-security.blogspot.de

Sourcecode-Analyse

Schwachstellen in Software gehören leider zum Alltag. Quelle der Schwachstellen sind Design- und Programmierfehler. Ein Weg, diese zu reduzieren, ist die Analyse des Quellcodes. Damit lassen sich viele Schwachstellen frühzeitig erkennen, egal, ob es sich um die Einhaltung von Standards, die Verwendung gefährlicher Codekonstrukte oder Logikfehler handelt.

Im Gegensatz zu einer Designanalyse liegt der Fokus bei der Sourcecode-Analyse auf der eigentlichen Implementierung. Dabei ersetzt die Analyse nicht das Testen, sondern dient als weitere Qualitätssicherungsmaßnahme.
Obwohl Sourcecode-Analysen, egal ob automatisiert oder manuell, schon lange zum Repertoire von Software-Entwicklern gehören, werden sie leider immer noch zu wenig eingesetzt. Jeder Fehler, der in der Entwicklung und nicht erst im Feld entdeckt wird, spart nämlich bares Geld.
Unsere Erfahrung aus dem Bereich der Produktzertifizierung zeigt klar, dass eine fokussierte Analyse selbst bei erfahrenen Entwicklern oft noch Schwachstellen im Code entdecken kann.
Es stellt sich also die Frage, wie nutzt man die Code-Analyse und wie wird sie in den Entwicklungszyklus eingebettet? Die Code-Analyse kann in verschiedenen Formen stattfinden, zum einen natürlich manuell, aber auch eine statische Code-Analyse mit entsprechenden Werkzeugen ist empfehlenswert.
Zusätzlich gibt es auch Werkzeuge, die Programme zur Laufzeit analysieren; der Fokus dieses Beitrags liegt aber auf der statischen Analyse.

Manuelle Analyse

Die manuelle Analyse gibt es als Code-Review innerhalb der Entwicklergruppe oder als externen Code-Audit schon lange. Das Prinzip „keine Integration ohne Prüfung“ ist ein Eckpfeiler des Qualitätsmanagements jeder Entwicklung. Allerdings ist das zusätzliche Lesen und Verstehen des Codes teuer, langwierig und fehleranfällig. Man braucht Entwickler, die das Programm und die Auswirkungen auf andere Teile so gut verstehen wie der Programmierer. Damit sind die Entwickler dann für ihre eigenen Projekte nicht verfügbar. Und je länger ein Review dauert, desto mehr lässt die Konzentration nach und Fehler werden übersehen.
Man muss also die manuelle Code-Analyse auf ein Minimum beschränken. Oft bedeutet das, dass nur die Änderungen im Code angesehen werden; das birgt die Gefahr, dass ungeänderte Teile, die man hätte ändern müssen, übersehen werden.
Deutlich effizienter ist es dagegen, zielgerichtet nach Sicherheits-Schwachstellen zu suchen und den dafür zu untersuchenden Code sauber abzugrenzen.

Eingrenzung

Ausgehend von einer Analyse der Architektur werden die Schnittstellen und Module bestimmt, welche für die detaillierte Untersuchung relevant sind.
Der Fokus liegt dabei auf den externen Schnittstellen, den für die Sicherheit relevanten Modulen und deren Interaktion. Typische Beispiele sind die Verarbeitung von Nutzereingaben in Anwender-Schnittstellen, die Kommunikation über einen Socket oder die APIs von Applikationen.
Ausgehend von den Schnittstellen wird die Verarbeitung der empfangenen, aber auch der zu sendenden Daten betrachtet. Im ersten Fall geht es um die Beeinflussung des Programms durch diese Daten, im zweiten um die Kontrolle über die abgegebenen Daten.
Ein anderer Ansatz ist der Fokus auf den Datenfluss sicherheitsrelevanter Daten. Wo werden kritische Daten (Passwörter, Berechtigungen) verarbeitet und gespeichert und wieder gelöscht? Auch diese Programmteile verdienen eine genaue Untersuchung.
Im nächsten Schritt werden Ressourcen identifiziert, die für die oben erkannten Module relevant sind. Das sind dann typischerweise auch jene Ressourcen, die für einen Angreifer interessant sind.
Als nächstes werden die Vertrauensverhältnisse ermittelt: Welche Komponente verlässt sich auf andere Komponenten des Systems bzw. auf externe Komponenten in der Umgebung? Welche Komponenten arbeiten in exponierten Bereichen?
Durch die zielgerichtete Eingrenzung wird zum einen der Arbeitsumfang reduziert, zum anderen kann man auch sicher sein, keine kritischen Bestandteile unbeachtet zu lassen.

Analyse

Nachdem die kritischen Komponenten, Ressourcen und Schnittstellen identifiziert wurden, kann nun die eigentliche Code-Analyse auf einer stark reduzierten Untermenge des Gesamtcodes stattfinden.

Beispielsweise werden dabei folgende Punkte betrachtet:

- Werden Eingaben vor der Verarbeitung geprüft und bereinigt? Erfolgt die Prüfung so, dass nicht bei der Prüfung schon Pufferüberläufe oder andere ungewollte Reaktionen erzeugt werden?

- Wie erfolgt die Prüfung von Privilegien beim Passieren von Schnittstellen zwischen verschiedenen Sicherheitsdomänen (z. B. an einer Systemcall-Schnittstelle)?
- Ist sichergestellt, dass alle Variablen immer initialisiert werden und erfolgt eine korrekte Initialisierung?
- Werden Credentials sofort explizit gelöscht, wenn sie nicht mehr benötigt werden? Passwörter und Schlüssel sollten auch im Speicher niemals länger als absolut notwendig vorgehalten werden.
- Werden bekannte, unsichere Bibliotheksfunktionen oder Funktionalitäten mit undefinierten Seiteneffekten verwendet (Speziell bei C/C++)?
- Werden Fehlercodes von Funktionen geprüft?

Statische Code-Analyse mit Werkzeugen

Im Gegensatz zur manuellen Analyse, die sehr zielgerichtet erfolgt, arbeiten Werkzeuge auf der gesamten Codebasis, bzw. in abgeschlossenen Modulen. Da ein Analyseprogramm weder Architektur- noch Designinformationen in die Analyse einbezieht, ist die Anzahl der „False Positives“, die dann erst durch manuelle Nachprüfung aussortiert werden müssen, typischerweise recht hoch. Dennoch rentiert es sich, solche Werkzeuge einzusetzen, denn es kommen immer noch genügend reale Probleme ans Licht, um den Overhead durch die Fehlalarme zu rechtfertigen.

Je nach Werkzeug gibt es dann auch Möglichkeiten, „False Positives“ im Code zu markieren, so dass diese Stellen bei einer erneuten Überprüfung nicht wieder anschlagen. Dies wird insbesondere dann wichtig, wenn ein Analysewerkzeug entwicklungsbegleitend eingesetzt wird. Einige der automatisierten Analysewerkzeuge lassen sich direkt in Eclipse oder andere Entwicklungsumgebungen integrieren und liefern schon beim Erstellen des Codes Rückmeldungen zu potentiellen Problemen.
Leider sind die automatisierten Werkzeuge nicht direkt vergleichbar. Sie haben sehr unterschiedliche Stärken und Schwächen; damit wird es auch schwer, konkrete Empfehlungen auszusprechen. In der Praxis ist ein Mix empfehlenswert. Hinzu kommt, dass die kommerziellen Werkzeuge mit sehr hohen Kosten zu Buche schlagen, die frei verfügbaren dagegen oft nicht sehr gut an die eigenen Bedürfnisse anpassbar sind. Als Startpunkt für die Auswahl ist *http://www.dwheeler.com/flawfinder/#othertools* empfehlenswert, hier sind praktisch alle bekannten Werkzeuge verzeichnet.

Ein interessanter Sonderfall für die statische Analyse ist Findbugs. Dieses Analyseprogramm für Java-Code verwendet nicht den Quellcode, sondern die JAR-Datei. Das ermöglicht dann auch die Analyse von Programmen bei denen der Quellcode nicht vorliegt. Umgekehrt bedeutet das aber auch, dass potentielle Angreifer diese Analyse an ausgelieferten JAR-Dateien vornehmen können. Damit wird die Analyse mit Findbugs praktisch zur Pflicht für alle Java Applikationen.

Fazit

In unserer Prüfstelle führen wir routinemäßig Code-Analysen durch. Dabei hat sich gezeigt, dass sie zu einer erhöhten Code-Qualität und zu einem höheren Sicherheitsniveau von Software führen.
Unsere Erfahrung zeigt aber, dass es auf lange Sicht besser ist, die Werkzeuge und auch manuelle Review-Prozesse in den Entwicklungsprozess zu integrieren, da das Feintuning zur Anpassung an die individuellen Gegebenheiten des Entwicklerteams Zeit braucht . Wir beraten daher bei der Auswahl der geeigneten Werkzeuge, helfen beim Training der Mitarbeiter und schieben mit den ersten Scans/ Reviews den Prozess in der Organisation an.

Werkzeugauswahl
http://www.dwheeler.com/flawfinder/#othertools

atsec NEWS

Kostenloser ISO/IEC 27001 Workshop

Im Oktober 2013 ist atsec der Allianz für Cybersicherheit des BSI (Bundesamt für Sicherheit in der Informationstechnik) als Mitglied beigetreten. Im Rahmen einer aktiveren Beteiligung als Partner der Allianz haben wir uns entschlossen, für Mitglieder der Allianz für Cybersicherheit einen kostenlosen Workshop zum Thema "Aufbau eines ISMS auf Basis von ISO 27001" durchzuführen.

Der Schwerpunkt liegt hierbei auf der Einführung des Standards bei kleinen und mittleren Unternehmen (KMUs). Für die praktische Umsetzung der ISO 27001 dient exemplarisch der Energie-Sektor, am Beispiel der Anwendung der ISO/IEC 27019:2013. Der Workshop ist aber grundsätzlich so gestaltet, dass er für alle Zuhörer, unabhängig von Firmengröße und Branche, eine Einführung in den Standard ISO 27001 bietet.

Für unsere Bestandskunden haben wir natürlich ebenfalls ein paar Plätze in diesem kostenlosen Workshop freigehalten. Sollten Sie sich also für die Implementierung eines Informationssicherheits-Managementsystems auf Basis ISO 27001 interessieren, eine Zertifizierung anstreben oder ganz allgemein Interesse an der Umsetzung dieses Standards haben, dann melden Sie sich doch einfach unter der Email-Adresse *workshop@atsec.de* zum Workshop an. Aufgrund einer maximalen Teilnehmerzahl von 30 Personen berücksichtigen wir Anmeldungen in Reihenfolge des Eingangs.

Wir würden uns sehr freuen, Sie bei unserem Workshop begrüßen zu dürfen.

Weitere Details finden Sie unter http://www.atsec.de/ aktuelles/workshop-isms- iso-27001/index.html

Details zur Veranstaltung

Veranstaltungsort: München;
genaue Adresse und Anreiseinformationen werden rechtzeitig an alle Teilnehmer verschickt
Veranstaltungsbeginn: 29.10.2014, 09:45 Uhr
Veranstaltungsende: 29.10.2014, ca. 16:30 Uhr

Agenda

09:45 - 11:00	■ Begrüßung ■ Gefährdungslage aktuell ■ Kurzeinführung „Informationssicherheits-Managementsystem" (ISMS) ■ Vergleich „ISO/IEC 27001" und „IT-Grundschutz auf Basis von ISO 27001"
Kurze Kaffeepause	
11:15 - 12:30	■ Wie implementiert man ein Informationssicherheits-Managementsystem nach ISO/IEC 27001? ■ Überblick über die ISO 27000 Standard-Familie ■ ISO 27001:2005 vs. ISO 27001:2013 ■ Abgrenzung: Informationssicherheit vs. IT Sicherheit vs. Datenschutz ■ Gültigkeitsbereich und Ausschlüsse ■ PDCA - Deming Cycle ■ Sicherheitsziele ■ Risikomanagement Informationssicherheit ■ Audits (Intern und Extern) ■ Was ist ein „Statement of Applicability"? ■ Softwareentwicklung nach ISO 27001 ■ Notfallplanung ■ Sicherheitsvorfälle ■ Dokumente und Dokumentenlenkung ■ Umsetzung von rechtlichen Anforderungen/Compliance ■ Verantwortung im Unternehmen für Informationssicherheit ■ Technische und organisatorische Maßnahmen
Mittagspause	
13:30 - 14:45	■ Praktische Umsetzung von Maßnahmen ■ ISO 27002 als „Best Practice" ■ Branchenspezifisch: DIN SPEC 27009 bzw. ISO/IEC 27019:2013 für die Energiewirtschaft **Projekt** ■ Projektablauf von der Implementierung bis zur (optionalen) Zertifizierung
Kaffeepause	
15:15 - 16:30	■ Überblick zur Integration mit anderen Management-Systemen ■ ISO 9001 ■ ISO 20000 ■ Datenschutz-Managementsystem
Fragenrunde	

IMPRESSUM

atsec information security GmbH
Steinstr. 70
81667 München
Deutschland
Telefon: +49-89-442-49-830
Telefax: +49-89-442-49-831
E-Mail: info@atsec.com

Vertretungsberechtigte Geschäftsführer:
Salvatore la Pietra
Staffan Persson
Registernummer: HRB 129439
Registergericht: Amtsgericht München
UST-ID-Nr. gemäß § 27a UStG: DE205370914
Verantwortlich für den Inhalt:
Staffan Persson (Anschrift s.o.)