HISOLUTIONS - SCHWACHSTELLENREPORT

Eine Analyse der identifizierten Schwachstellen in Penetrationstests des Jahres 2016

MOTIVATION

HiSolutions führt jedes Jahr eine große Anzahl von unterschiedlichen Penetrations- und Schwachstellentests durch. Immer wieder werden wir dabei gefragt, wie die Ergebnisse des einzelnen Tests gegenüber „typischen“ Ergebnissen einzustufen sind, und ob die identifizierten Probleme bei anderen Unternehmen ähnlich bestehen.

Wir haben diese Fragen zum Anlass genommen, die von uns in den letzten Jahren durchgeführten Tests jahresweise auszuwerten und die jeweils identifizierten Schwachstellen in Kategorien zusammenzufassen. Diese Aggregation erlaubt uns, einerseits die Vertraulichkeit der Projektergebnisse gegenüber unseren Kunden zu wahren, andererseits aber Aussagen abzuleiten über typische Testergebnisse und besondere Problembereiche, die entweder besonders häufig auftauchen oder besonders schwerwiegende Lücken darstellen. Durch die Fortschreibung der Auswertung über die Jahre hinweg können dabei auch Trends und Entwicklungen in der Sicherheitslage deutlich werden.

HISOLUTIONS

VORGEHEN

Dieser Report beruht auf einer Auswertung der Ergebnisse aus insgesamt 43 Penetrations- und Schwachstellentests, die im Jahr 2016 durchgeführt wurden. Die Tests betreffen verschiedene Zielumgebungen, von Netzwerkinfrastrukturen über Web-Anwendungen bis hin zu einzelnen Systemen und Verfahren, sind also nicht direkt miteinander vergleichbar. Durch die Kategorienbildung bei den Schwachstellen lassen sich dennoch interessante Beobachtungen ableiten.

Für die Kategorien haben wir uns zunächst an den „OWASP Top 10“ orientiert. Diese Veröffentlichung des OWASP-Projektes aus dem Jahr 2013[1] umfasst eine Systematik der schwerwiegendsten Schwachstellen *für Web-Anwendungen*, die dort auf der Grundlage einer Berechnung der Schweregrade auf der Basis von Häufigkeiten und Auswirkungen erstellt wurde. Die Kategorien lassen sich dabei z. T. auch auf andere Testziele gut übertragen, decken jedoch nicht alle unsere Befunde vollständig ab, so dass wir einige eigene Kategorien ergänzt haben.

Die Aggregation bringt einige praktische Schwierigkeiten mit sich: Wegen der Unterschiedlichkeit der durchgeführten Tests ließen sich keine relevanten Aussagen zur Häufigkeit einer Schwachstelle pro System oder Anwendung ermitteln. Auch fassen wir in den Projektberichten gleichartige Schwachstellen auf verschiedenen Systemen häufig zu einem Befund zusammen, so dass eine Zählung der Befunde hier ebenfalls nur begrenzte Aussagekraft hat. Wir haben uns daher entschlossen, als Maß die Häufigkeit des Auftretens eines Schwachstellentyps pro Projekt anzusetzen. Dadurch wird deutlich, welchen Schwachstellen wir in unterschiedlichen Projekten besonders häufig begegnen, und welche eher selten oder nur in besonderen Zielumgebungen auftauchen.

Für die Bewertung der Relevanz einer Schwachstelle verwenden wir in unseren Prüfberichten ein standardisiertes Schema, in dem wir aus der Bewertung der Komplexität des Angriffs und des zu erwartenden Schadens zu einer Einordnung in die folgenden Kategorien kommen:

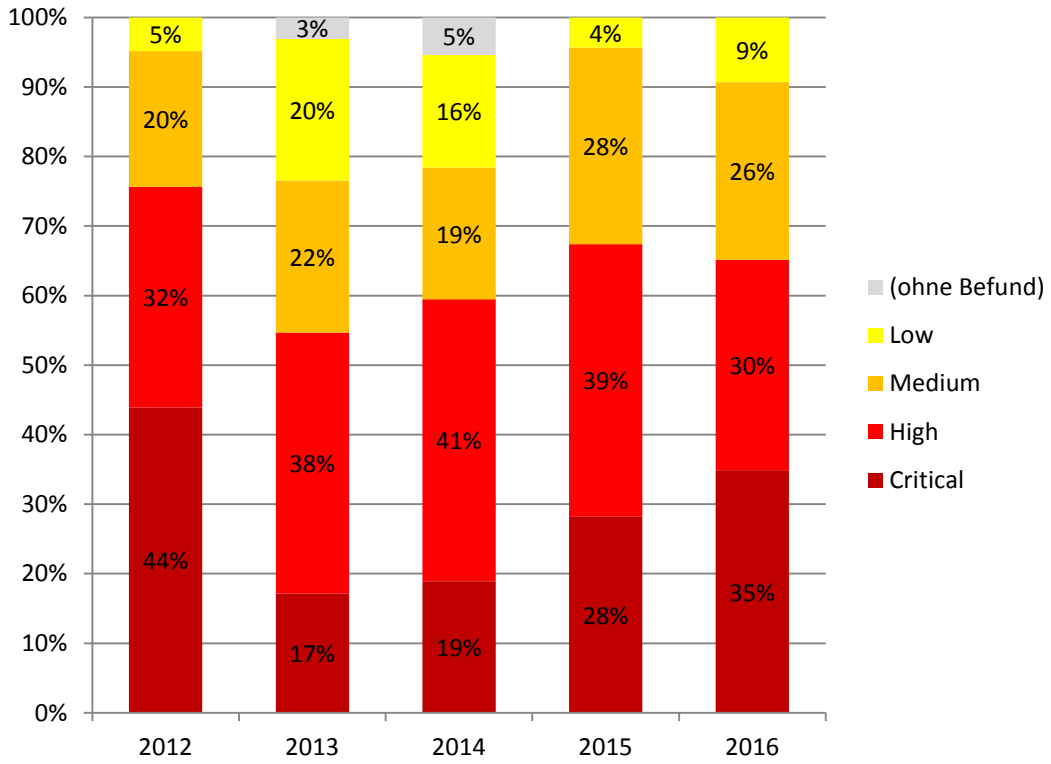
CRITICAL (C)	Die getesteten Systeme sind akut gefährdet, umgehendes Handeln ist (in der Regel noch während der Testdurchführung) erforderlich.
HIGH (H)	Die Schwachstelle hat eine hohe praktische Relevanz und sollte priorisiert behoben werden.
MEDIUM (M)	Die Schwachstelle besitzt ein relevantes Schadenspotenzial, dieses kann aber nur in bestimmten Umständen oder in Verbindung mit anderen Problemen realisiert werden.
LOW (L)	Die Schwachstelle stellt für sich keine unmittelbare Gefahr dar, kann jedoch Angriffe über andere Schwachstellen erleichtern oder verstärken.

Rein informative Befunde (z. B. festgestellte funktionale Fehler ohne Sicherheitsbezug) wurden in der Zählung nicht berücksichtigt.

Zusätzlich haben wir die Befunde mit den Ergebnissen unserer Schwachstellenreports von 2012 bis 2015 verglichen.

[1] https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Auch wenn sich im Vergleich der letzten Jahre immer wieder leichte Verschiebungen ergeben haben, bleibt doch die Befundlage über die Jahre insgesamt relativ gleich. Auch 2016 hat dabei der Anteil von kritischen Schwachstellen in unseren Tests wieder deutlich zugenommen:



Maximale Kritikalität in Untersuchungen der letzten fünf Jahre

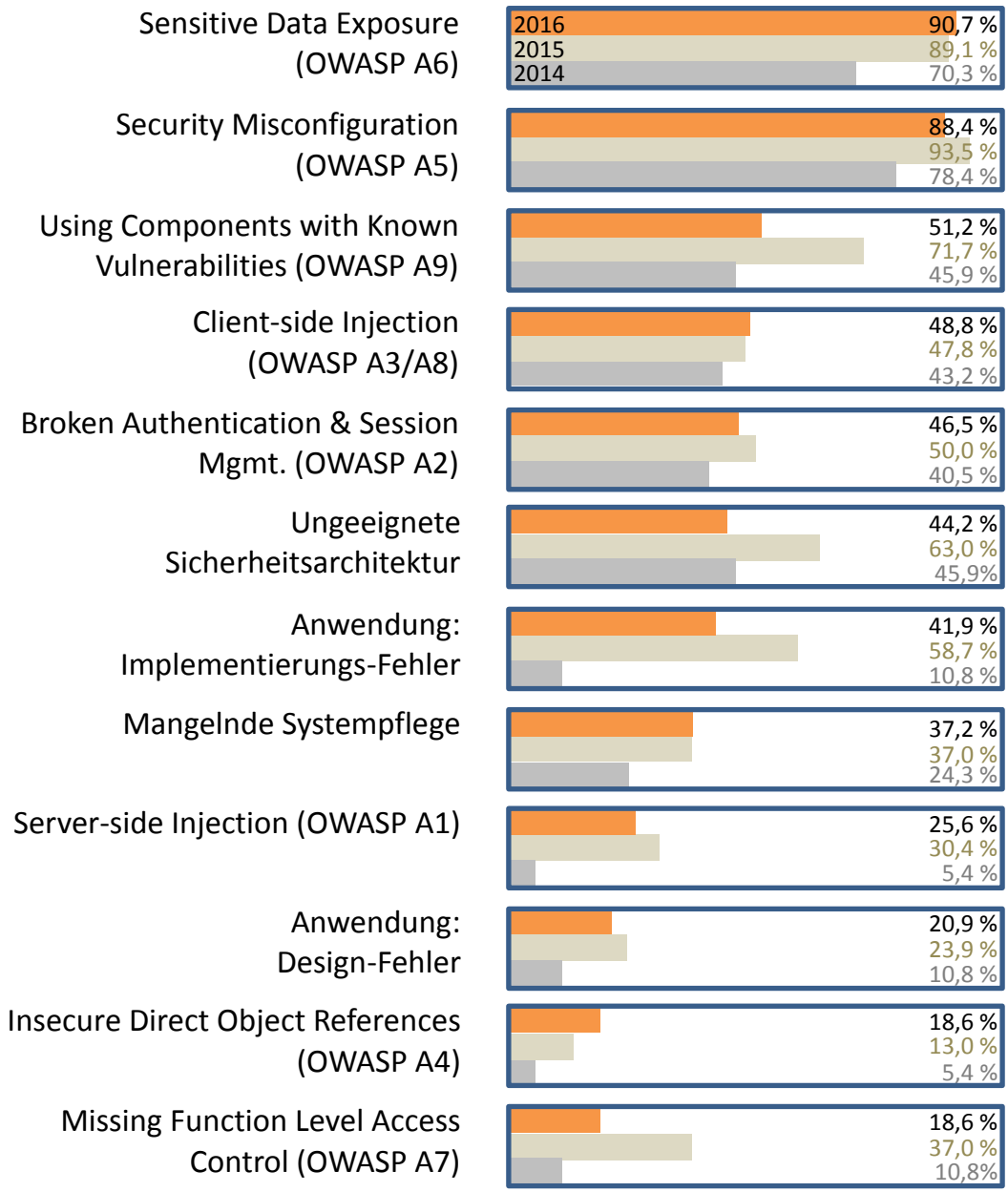
Wie schon im Vorjahr konnten wir kein Projekt ohne sicherheitsrelevanten Befund abschließen. Der Anteil von Tests mit schweren und kritischen Befunden ist geringfügig gesunken, bewegt sich aber weiterhin bei nahezu zwei Dritteln (65 %). Sorge bereitet allerdings die erneute deutliche Zunahme bei Tests mit kritischen Befunden (35 % gegenüber 28 % im Vorjahr).

Analysiert man die gefundenen Schwachstellen thematisch, ergibt sich folgendes Bild:

Die Gruppe der Schwachstellen mit der häufigsten Einstufung als „kritisch“ umfasst den Einsatz von veralteten oder bekanntermaßen unsicheren Komponenten (OWASP A9), gefolgt von sicherheitsrelevanten Konfigurationsfehlern (OWASP A5), angreifbaren Architekturen und serverseitigen Injection-Angriffen (OWASP A1). Kritische Probleme durch eine fehlende Zugriffskontrolle auf funktionaler Ebene (OWASP A7) haben gegenüber dem Vorjahr deutlich abgenommen.

Insgesamt findet sich weiterhin eine breite Mischung aus Problemen in der Entwicklung, der Konfiguration und dem Betrieb von IT-Infrastrukturen.

Die folgende Grafik zeigt die von uns definierten Schwachstellenkategorien jeweils mit der Häufigkeit ihres Auftretens im Projekt, verglichen mit den beiden Vorjahren:



An der Spitze verzeichnen wir auffällig häufig Fälle von unzureichend geschützten Informationen (OWASP A6) und Konfigurationsfehlern (OWASP A5), die jeweils wie schon im Vorjahr in ungefähr 9 von 10 Projekten aufgetreten sind. Dies sind typischerweise Fehler, die nicht aus der Entwicklung der Anwendungen, sondern aus dem Betrieb der Systeme resultieren. Bei der Härtung der Systemkonfiguration von Serversystemen besteht also durchgängig weiter Handlungsbedarf.

Auch das Patchmanagement, das eigentlich sicherstellen müsste, dass Softwareversionen mit bekannten Schwachstellen aktualisiert werden, bleibt eine Herausforderung, die nur wenige Unternehmen durchgängig meistern. In mehr als der Hälfte unserer Tests haben wir veraltete Software mit bekannten Schwachstellen im Betrieb gefunden.

Fehler aus der Anwendungsentwicklung wie Injection-Angriffe oder Fehler beim Session Handling bleiben über die Jahre auf einem hohen Niveau. Der vor einigen Jahren noch prognostizierte Rückgang solcher Probleme durch den Einsatz ausgereifter Frameworks ist in der Praxis nicht eingetreten.

Wieder in relativ großer Häufigkeit finden sich die Kategorien „Ungeeignete Sicherheitsarchitektur“ und „Implementierungsfehler“, beide mit Häufigkeiten über 40 %. Hier bestätigt sich unsere Erfahrung aus Sicherheitsaudits, dass Sicherheitsfragen in IT-Projekten in vielen Fällen erst im Zuge der Betriebseinführung und damit sehr spät im Projektverlauf betrachtet werden. Probleme aus den beiden genannten Fehlerkategorien entstehen aber typischerweise viel früher im Projekt.

Ebenfalls häufig bleiben Schwachstellen in der Kategorie „Mangelnde Systempflege“. Die Kategorie umfasst dabei nicht veraltete Software, die wir ja in der eigenen Kategorie OWASP A9 führen. Hier sind andere Probleme der Pflege von Systemen zusammengefasst, z. B. der Einsatz von Debugging- und Testfunktionen auf Produktivsystemen, veraltete Konfigurationseinstellungen, nicht benötigte Dienste und vergleichbare Nachlässigkeiten. Wie schon in den Vorjahren sind die damit verbundenen Sicherheitsprobleme aber nur selten schwerwiegend, wenngleich im Trend auch hier eine Zunahme des Schweregrads erkennbar ist.

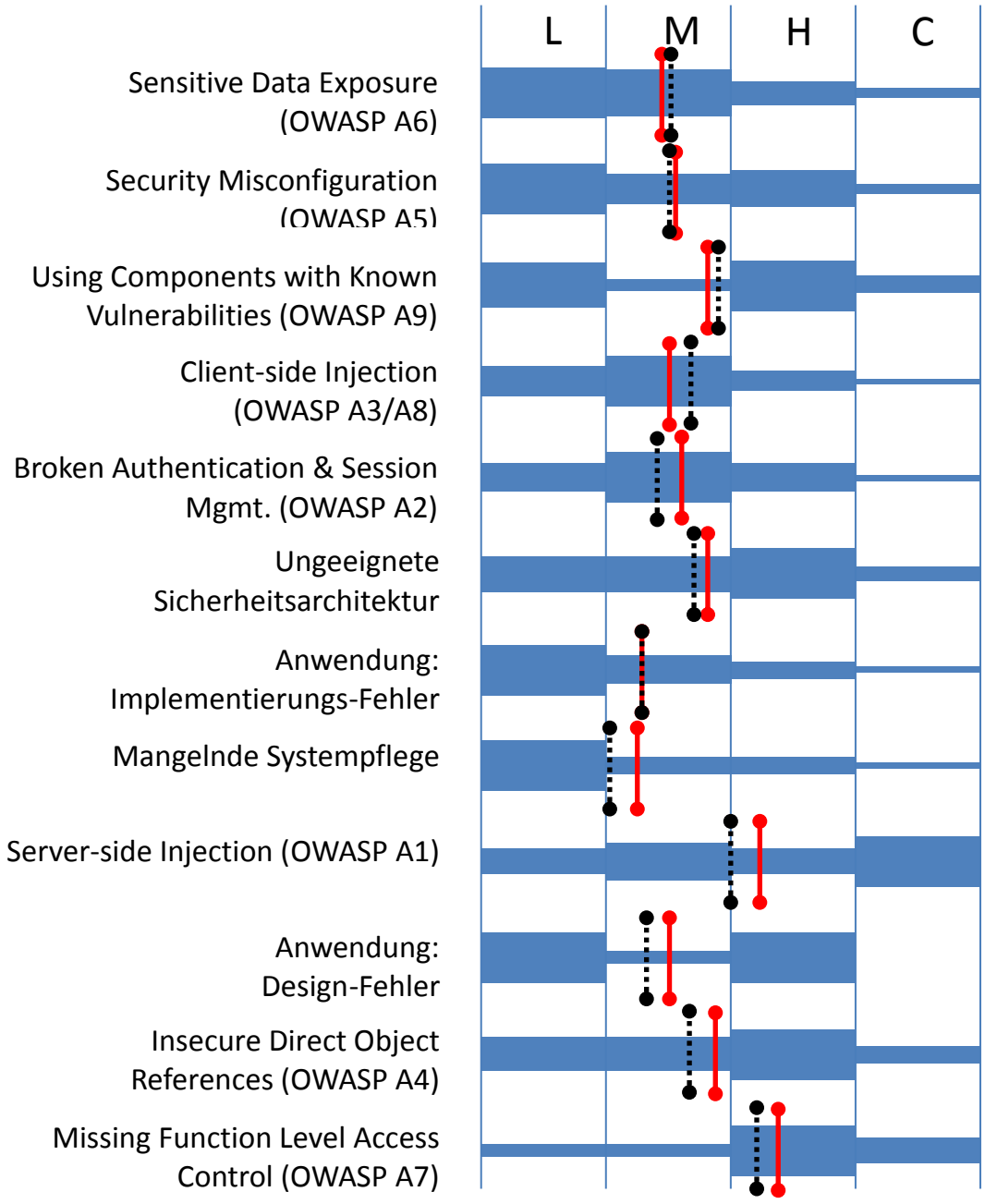
Bei den serverseitigen Injection-Angriffen hatten wir schon im Vorjahr die Vermutung ausgesprochen, dass das äußerst seltene Auftreten dieser Fehlerklasse im Jahr 2014 ein statistischer „Ausreißer“ nach unten war. Dies scheint sich zu bestätigen – auch 2016 war die Fehlerklasse in knapp über 25 % der Prüfungen wieder deutlich vertreten. Wie schon in den Vorjahren hat diese Fehlerklasse den höchsten Anteil kritischer Befunde: Wenn der Angreifer eigenen Code auf Serversystemen ausführen kann, ist das Schadenspotenzial meistens beträchtlich – vom Auslesen und der Manipulation der Datenbestände bis hin zu massiven Störungen des Betriebs.

Auf den nächsten drei Plätzen folgen wieder Fehlerklassen, die bereits Design-Entscheidungen oder fehlende Programmiervorgaben im Entwicklungsprojekt aufzeigen. Auch hier könnte durch die konsequente Einbindung von Security-Fachleuten in IT-Projekte vermutlich ein substanzieller Rückgang erzielt werden. In allen drei Kategorien hat der durchschnittliche Schweregrad der Befunde im letzten Jahr zugenommen. Immerhin sind fehlerhafte Design-Entscheidungen die einzige Kategorie ohne kritische Befunde – es wäre auch ein schlechtes Zeichen, wenn solche schwerwiegenden Fehler in den Projekten gar nicht erkannt würden. Die Auswirkungen von Design-Fehlern sind im Mittel aber höher als bei den Implementierungsfehlern, was die Notwendigkeit einer frühen Einbindung der Security nochmals belegt.

Wo wir gegenüber dem Vorjahr Rückgänge im Auftreten von Fehlerarten festgestellt haben, scheinen diese aber eher auf Ausreißer in den Prüfungen aus dem Jahr 2015 hinzudeuten. Gegenüber 2014 liegt auch hier meistens eine deutliche Zunahme vor.

Zusammenfassend bleiben die Ergebnisse der Tests besorgniserregend. Die erhöhte Aufmerksamkeit der Öffentlichkeit für IT-Sicherheitsthemen hat weiterhin kaum zu erkennbaren Verbesserungen in den tatsächlich betriebenen IT-Infrastrukturen geführt.

In der folgenden Grafik zeigen wir die vorgenommenen Einstufungen als blaue Balken, die die relative Verteilung der Zuordnung abbilden. Der durchschnittlich gewählte Schweregrad ist als rote Markierung eingezeichnet. Die gestrichelten schwarzen Linien zeigen die Durchschnittswerte aus dem Vorjahr:



Lesebeispiel: In der Kategorie “Using Components with Known Vulnerabilities (OWASP A9)” reichten die vorgenommenen Einstufungen von „low“ bis „critical“; der durchschnittliche Schweregrad war etwas über „medium“, bei einem leichten Rückgang gegenüber dem Vorjahr. Die einzelnen Befunde waren in den meisten Fällen entweder schwerwiegend (high) oder gering (low).

Die große Bandbreite bei der Bewertung der Schwachstellen innerhalb der einzelnen Kategorien zeigt, dass die Einstufung je nach Art des konkreten Befunds, insbesondere aber auch der Kritikalität der betroffenen Systeme und Daten, individuell zu ermitteln ist.

SENSITIVE DATA EXPOSURE (OWASP A6)

Unter diese Kategorie fallen alle Schwachstellen, die zu einem mangelhaften Schutz sensibler Daten führen. Dazu gehören neben einer fehlerhaften Konfiguration der Transportsicherheit (SSL) auch ein mangelhafter Schutz von Passwörtern und anderen sensiblen Daten durch eine fehlende Verschlüsselung oder den Einsatz veralteter Verschlüsselungsverfahren. Gerade die Anwendung kryptografischer Algorithmen hält viele Fallstricke bereit, die von Angreifern ausgenutzt werden können. Allerdings sind solche Schwachstellen nur selten als kritisch zu bewerten, da sie zumeist nur unter bestimmten Umständen oder mit einem erheblichen Aufwand ausnutzbar sind.

SECURITY MISCONFIGURATION (OWASP A5)

Diese Kategorie umfasst alle Arten von Konfigurationseinstellungen, die zu Schwachstellen oder Angriffspunkten führen, und ist daher in sich sehr heterogen – wir haben über 60 verschiedene Arten von Befunden dieser Kategorie zugeordnet. Viele der Konfigurationsprobleme führen jedoch auch nur zu geringen Risiken, so dass die meisten Einstufungen hier niedrig bis mittel ausgefallen sind. Kritisch sind lediglich bestimmte Fälle der Preisgabe von Informationen über technischen Konfigurationsdaten, Directory-Listings oder nicht gelöschte Beispiel- und Hilfedateien, die in den entsprechenden Fällen jeweils einen unmittelbaren Ansatzpunkt für Angriffe gegeben haben.

USING COMPONENTS WITH KNOWN VULNERABILITIES (OWASP A9)

Unter diese Kategorie fällt eine Vielzahl von Schwachstellen, die insbesondere aus einem mangelhaften Software- und Patchmanagement resultieren: Veraltete Software, vom Betriebssystem über die Anwendungsserver, Frameworks, die Anwendungssoftware und Erweiterungen oder Plug-Ins, kann eine Vielzahl von Schwachstellen beinhalten, die nach der Veröffentlichung leicht von Angreifern ausgenutzt werden können. Wird die Software nicht sorgfältig gepflegt, können schnell Lücken entstehen, die ein großes Schadenspotenzial beinhalten.

CLIENT-SIDE INJECTION (OWASP A3/A8)

Clientseitige Injection-Angriffe basieren auf dem Prinzip, dass der Angreifer in die Anwendung Programmcode einbringt, der auf dem Client eines Anwenders ungewollt zur Ausführung gelangt. In dieser Kategorie wurden OWASP A3 (Cross-Site Scripting, XSS) und OWASP A8 (Cross-Site Request Forgery, CSRF) zusammengefasst, da letztere eher selten und in der Regel in Verbindung mit ersterer auftritt. XSS macht hier sowohl bezüglich des Auftretens als auch der Schwere den Löwenanteil aus (ca. 60 %, über 90 % der hohen und kritischen Bewertungen), wobei es meist um reflektiertes (also dem Anwender über einen Link untergeschobenes), vereinzelt auch um persistentes XSS (dauerhaft in die Anwendung eingebrachten Schadcode) geht.

BROKEN AUTHENTICATION & SESSION MGMT. (OWASP A2)

Unter diese Kategorie fällt eine Vielzahl von Schwachstellen, die mit unterschiedlicher Häufigkeit und Kritikalität vorzufinden sind (wir haben über 30 verschiedene Arten von Einzelbefunden identifizieren können). Besonders schwerwiegend sind Session-Tokens in URLs, Session-Fixation-Angriffe sowie in bestimmten Fällen mangelhaft geschützte Session-Cookies, unsichere SSH-Schlüssel, zu wenig Entropie in Session-IDs oder in Einzelfällen Logins mit Default-Credentials oder gar ohne jede Zugangskontrolle.

UNGEEIGNETE SICHERHEITSARCHITEKTUR

Relativ häufig sind wir in unseren Projekten auf Sicherheitsarchitekturen gestoßen, die ihre Schutzfunktion nicht erfüllen. Dies begründet sich manchmal in fehlenden Schutzmechanismen (Firewalls), z. T. jedoch auch in vorhandenen, aber in der vorliegenden Konfiguration nicht wirksamen Sicherheitssystemen. Dieser Effekt ist besonders bei sogenannten Web Application Firewalls (WAF) häufiger zu beobachten. Ebenfalls in diese Kategorie haben wir eine aus Sicherheitssicht unzureichende Trennung von Produktiv- und Testumgebungen gezählt.

ANWENDUNG: IMPLEMENTIERUNGS-FEHLER

Wie bei der Vielzahl und Vielfalt existierender Anwendungen zu erwarten, bilden die hier zusammengefassten gut zwei Dutzend Schwachstellen einen bunten Strauß an Dingen, die bei der Implementierung von Anwendungen falsch gemacht oder vergessen wurden – über die von den OWASP-Kategorien bereits erfassten Fehlermöglichkeiten hinaus. Besonders kritische Fälle stehen oft im Zusammenhang mit mangelnder Rechteprüfung beim Lesen oder Schreiben sowie beim Upload von Dateien.

MANGELNDE SYSTEMPFLEGE

In einigen Tests stießen wir auf Umgebungen, die durch mangelnde Systempflege eine unnötig große Angriffsfläche boten, z. B. durch den Weiterbetrieb ungenutzter Systeme und Anwendungen oder Test- und Beispielanwendungen. Soweit solche Szenarien nicht zu unmittelbar ausnutzbaren Schwachstellen führten (und dann den entsprechenden anderen Kategorien zugeordnet wurden), wurden sie hier zusammengefasst.

SERVER-SIDE INJECTION (OWASP A1)

Ähnlich wie bei den clientseitigen Injection-Angriffen bringt auch bei der serverseitigen Injection der Angreifer eigenen Programmcode in die Anwendung ein, der hier jedoch auf der Serverseite ausgeführt wird und dadurch ein besonders hohes Schadenspotenzial hat. Der nach wie vor überwiegende Anteil besteht dabei in SQL-Injections, bei denen der Angreifer Datenbankabfragen der Anwendung manipuliert und sich so unbefugten Zugriff auf Daten und Funktionen verschafft. Diese Angriffe stufen wir oftmals als kritisch ein, im letzten Untersuchungszeitraum waren jedoch nur solche Schwachstellen auszumachen, deren Schadenspotenzial beschränkt war. In weiteren Fällen sind verschiedene andere Arten von Code-Injection-Lücken vorzufinden.

ANWENDUNG: DESIGN-FEHLER

Designfehler in Anwendungen sind zum Glück selten, dann aber oftmals gefährlich. Die Ausprägungen sind unterschiedlich, Beispiele sind Datenbankzugriff mit administrativen Rechten, Zulassen trivialer Passwörter, unnötige Exportfunktionen, unsichere Schnittstellen oder die ungewollte Preisgabe von Nutzerinformationen.

INSECURE DIRECT OBJECT REFERENCES (OWASP A4)

Der Zugriff auf Ressourcen sollte immer durch eine entsprechende Berechtigungsprüfung abgesichert werden. Wir treffen jedoch regelmäßig auf Fälle, in denen Daten über Parameter, z. B. als URL-Bestandteil, direkt referenziert werden, ohne dass eine ausreichende Rechteprüfung vorgenommen wird. Durch Änderung der Parameter (z. B. einfaches „Hochzählen“ oder durch Verständnis des entsprechenden Nummernschemas) kann dann auf andere Datenobjekte zugegriffen werden, für die keine Berechtigung vorliegt.

MISSING FUNCTION LEVEL ACCESS CONTROL (OWASP A7)

Diese Kategorie umfasst Schwachstellen durch URLs, die vor unbefugtem Zugriff nicht ausreichend geschützt sind, i. d. R. weil der Anwendungsentwickler einen direkten Aufruf der URL durch einen Angreifer nicht erwartet. Dabei werden sensible Daten oder Anwendungsfunktionen ohne Authentifizierung oder für nicht berechtigte Nutzer zugänglich gemacht. Schwerere Schwachstellen in diesem Bereich betreffen vor allem bestimmte administrative Logins, die nicht von außen erreichbar sein sollten, oder administrative Bereiche, die völlig ohne Authentifizierung erreichbar sind, sowie besonders sensible Kundendaten und -dokumente.

FAZIT

Auch im fünften Jahr unserer Erhebung geben die Ergebnisse ein ähnliches Bild ab wie in den Vorjahren. Dies bestätigt uns, dass sich aus dem Gesamtbild der doch begrenzten Anzahl ausgewerteter Projekte durchaus Aussagen ableiten lassen, denen eine übergreifende Gültigkeit zukommt.

Dabei steigt der Anteil der Tests mit kritischen Schwachstellen in den letzten Jahren immer weiter an – mit mittlerweile 35 % gibt er ein unerfreuliches Bild der IT-Sicherheit der getesteten Umgebungen. Von drei Penetrationstests liefert einer eine kritische, ein weiterer noch eine schwerwiegende Schwachstelle. In keinem einzigen Test war die geprüfte Umgebung frei von Sicherheitsmängeln.

Die häufigsten Fehler finden sich bei unsicheren Konfigurationseinstellungen von Diensten und Systemen und bei der unbeabsichtigten Offenlegung sensibler Informationen, also durch Schwachstellen mit Ursachen in der IT-Betriebsführung. Andere Problemursachen treten für sich deutlich weniger häufig auf, haben dafür aber zum Teil schlimmere Folgen.

Bei den Auswirkungen der gefundenen Schwachstellen gibt es zwei deutliche Spitzenreiter: Fehlende Berechtigungsprüfungen und serverseitige Injection-Angriffe. Im letzteren Fall war das Auftreten dieser Schwachstelle sogar in den meisten Fällen direkt mit einem kritischen Problem verbunden, das unmittelbares Handeln erforderte. Beide Fehlerarten führen meistens dazu, dass ein Angreifer weitreichenden unbefugten Zugriff auf Daten und Funktionen erhält und dadurch große Schäden verursachen kann.

Nach wie vor verzeichnen wir eine breite Streuung möglicher Probleme. Der Varianten-reichtum der in der Praxis vorgefundenen Schwachstellen erschwert ein einfaches und schnelles Auffinden beispielsweise durch automatisierte Verfahren. Ein „Durchtesten“ ausgewählter „Top-5“- oder „Top-10“-Lücken erweist sich weiterhin als nicht hinreichend.

Regelmäßige Penetrationstests helfen, die vorhandenen Probleme zu identifizieren und abzustellen. Die Aufrechterhaltung eines angemessenen Sicherheitsniveaus gelingt am besten in der Kombination einer systematischen Risikoanalyse und der Verifikation der Wirksamkeit der umgesetzten Maßnahmen im praktischen Test.

KONTAKT

Frank Rustemeyer
Director System Security
Fon +49 30 533 289-0
rustemeyer@hisolutions.com

HiSolutions AG
Bouchéstraße 12
12435 Berlin

info@hisolutions.com
www.hisolutions.com
Fon +49 30 533 289 0
Fax + 49 30 533 289 900