APP.4.7 Zeitreihen-Datenbanken

1. Beschreibung

1.1. Einleitung

Zeitreihen-Datenbanken (Time-Series Databases, TSDB) sind hochspezialisierte Datenbanksysteme, die für das Speichern, Verwalten und Auswerten von Daten mit einem Zeitstempel optimiert sind. Systeme wie Prometheus, InfluxDB oder TimescaleDB bilden das Fundament für modernes IT-Monitoring, die Analyse von Anwendungs-Metriken und die Verarbeitung von Daten aus dem Internet der Dinge (IoT).
Ihre Architektur ist auf extrem hohe Schreiblasten und schnelle, aggregierte Abfragen über Zeiträume ausgelegt. Dies führt zu spezifischen Betriebsmodellen, wie dem "Scraping" von Metrik-Endpunkten (Prometheus) oder der Entgegennahme von Datenströmen von unzähligen Sensoren und Agenten. Diese Besonderheiten bringen eigene Sicherheitsherausforderungen mit sich: ungeschützte HTTP-Endpunkte, die sensible Systeminformationen preisgeben, das Risiko der Datenmanipulation durch kompromittierte Agenten und die Notwendigkeit, riesige Datenmengen über lange Zeiträume zu verwalten und zu schützen.

1.2. Zielsetzung

Dieser Baustein zeigt einen systematischen Weg auf, um Zeitreihen-Datenbanken und die zugehörigen Ökosysteme aus Agenten und Endpunkten sicher zu konfigurieren und zu betreiben. Ziel ist es, die Integrität der erfassten Messwerte zu gewährleisten, die Vertraulichkeit der oft sensiblen Monitoring-Daten zu schützen und die Verfügbarkeit der zentralen Monitoring-Infrastruktur sicherzustellen.

1.3. Abgrenzung und Modellierung

Der Baustein APP.4.7 ist auf alle eingesetzten Zeitreihen-Datenbanken anzuwenden. Dieser Baustein schließt eine Lücke im IT-Grundschutz-Kompendium, das bisher primär relationale Datenbanken (siehe APP.4.3) adressiert. Er behandelt nicht die grundlegende Absicherung des Servers (siehe OPS.1.1-Bausteine) oder die sichere Entwicklung der Anwendungen, die Metriken bereitstellen (siehe CON.1). Der Fokus liegt auf der sicheren Konfiguration der Datenbank selbst und der typischen Interaktionsmuster im Zeitreihen-Umfeld.

2. Gefährdungslage

Für den Baustein APP.4.7 sind folgende spezifische Bedrohungen und Schwachstellen von besonderer Bedeutung:

Manipulation von Messdaten

Gelingt es einem Angreifer, gefälschte Metrik-Daten in die Datenbank zu schreiben, kann er die Überwachung der IT-Systeme untergraben. So könnten Angriffe verschleiert werden, indem Leistungsdaten (z. B. CPU-Auslastung) künstlich niedrig gehalten werden. Ebenso

können gezielt Fehlalarme ausgelöst werden, um von einem realen Vorfall abzulenken oder das Betriebspersonal zu ermüden.

Unautorisiertes Auslesen von Monitoring-Daten

Monitoring-Daten geben tiefe Einblicke in die Infrastruktur, die Auslastung von Anwendungen und teilweise sogar in Geschäftsmetriken. Ungeschützte Abfrage-APIs ermöglichen es Angreifern, wertvolle Informationen für die Vorbereitung weiterer Angriffe zu sammeln, indem sie Schwachstellen oder wenig genutzte Systeme identifizieren.

Offene und ungeschützte Metrik-Endpunkte

Anwendungen und Systeme, die von pull-basierten TSDBs wie Prometheus überwacht werden, stellen ihre Metriken oft über ungeschützte HTTP-Endpunkte bereit. Sind diese aus nicht vertrauenswürdigen Netzwerken erreichbar, kann ein Angreifer detaillierte Informationen über Softwareversionen, Abhängigkeiten und die interne Funktionsweise der Anwendung auslesen.

Denial-of-Service durch Daten- oder Abfrageflut

Zeitreihen-Datenbanken sind anfällig für Angriffe, die auf die Erschöpfung von Ressourcen abzielen. Ein Angreifer könnte durch das Senden von Daten mit extrem hoher Varianz (hohe Kardinalität) den Speicher und die CPU der Datenbank überlasten. Ebenso können komplexe, langlaufende Abfragen das System lahmlegen und so die gesamte Überwachungsinfrastruktur zum Ausfall bringen.

Kompromittierung über unsichere Web-Schnittstellen

Viele Zeitreihen-Datenbanken werden mit integrierten Web-Oberflächen und mächtigen HTTP-APIs für Konfiguration und Abfrage ausgeliefert. Sind diese Schnittstellen nicht ausreichend gehärtet (z. B. durch fehlende Authentifizierung, schwache Standardpasswörter oder Cross-Site-Scripting-Schwachstellen), können sie einem Angreifer als Einfallstor zur Übernahme der Datenbank dienen.

3. Anforderungen

Im Folgenden sind die spezifischen Anforderungen des Bausteins APP.4.7 aufgeführt.

3.1. Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für diesen Baustein vorrangig erfüllt werden.

APP.4.7.A1 Absicherung des Netzwerkzugriffs auf die Datenbank (B)

Der Netzwerkzugriff auf die Zeitreihen-Datenbank MUSS auf die Systeme beschränkt werden, die Daten schreiben oder lesen müssen. Die Datenbank MUSS durch eine Firewall geschützt werden. Dienste SOLLTEN nur an vertrauenswürdige Netzwerkschnittstellen gebunden werden.

APP.4.7.A2 Aktivierung der Authentifizierung für Schreib- und Lesezugriffe (B)

Alle Schreib- und Lesezugriffe auf die Datenbank MÜSSEN authentifiziert werden. Es MÜSSEN sichere Passwörter oder Authentifizierungs-Token verwendet werden. Ein anonymer Zugriff MUSS deaktiviert werden.

APP.4.7.A3 Absicherung der administrativen Schnittstellen (B)

Der Zugriff auf administrative Web-Oberflächen und Konfigurations-APIs MUSS durch starke, nicht standardisierte Anmeldedaten geschützt sein. Standard-Benutzerkonten MÜSSEN umbenannt oder deaktiviert werden.

APP.4.7.A4 Schutz von Metrik-Endpunkten (B)

Metrik-Endpunkte, die von pull-basierten Systemen (z. B. Prometheus) abgefragt werden, MÜSSEN so konfiguriert sein, dass sie nur für das Monitoring-System erreichbar sind. Dies MUSS durch Netzsegmentierung oder Firewall-Regeln sichergestellt werden.

3.2. Standard-Anforderungen

Gemeinsam mit den Basis-Anforderungen entsprechen die folgenden Anforderungen dem Stand der Technik. Sie SOLLTEN grundsätzlich erfüllt werden.

APP.4.7.A5 Verschlüsselung der gesamten Kommunikation (S)

Die gesamte Kommunikation mit der Zeitreihen-Datenbank SOLLTE mittels TLS verschlüsselt werden. Dies umfasst die Verbindung von Agenten und Sensoren zur Datenbank, die Abfragen von Benutzern und Dashboards sowie die Kommunikation zwischen den Knoten in einem Cluster.

APP.4.7.A6 Umsetzung eines differenzierten Berechtigungskonzepts (S)

Es SOLLTE ein Berechtigungskonzept nach dem Prinzip der geringsten Rechte umgesetzt werden. Schreibende Agenten SOLLTEN nur Berechtigungen für die von ihnen benötigten Metriken erhalten. Lesende Benutzer (z. B. für Dashboards) SOLLTEN nur auf die für sie relevanten Daten zugreifen können.

APP.4.7.A7 Systematische Verwaltung von Datenaufbewahrungszeiten (S)

Es SOLLTEN Richtlinien für die Aufbewahrung von Daten (Retention Policies) definiert und technisch umgesetzt werden. Um unkontrolliertes Speicherwachstum zu verhindern, SOLLTEN veraltete Daten nach einer definierten Frist automatisch gelöscht oder aggregiert werden (Downsampling).

APP.4.7.A8 Protokollierung und Überwachung sicherheitsrelevanter Ereignisse (S)

Administrative Änderungen, fehlgeschlagene Anmeldeversuche und kritische Systemereignisse der Datenbank SOLLTEN protokolliert und an ein zentrales Log-Management übergeben werden. Die Protokolle SOLLTEN auf Anomalien überwacht werden.

3.3. Anforderungen bei erhöhtem Schutzbedarf

Die folgenden Anforderungen sind exemplarische Vorschläge für ein Schutzniveau, das über den Stand der Technik hinausgeht. Sie SOLLTEN bei erhöhtem Schutzbedarf in Betracht gezogen werden.

APP.4.7.A9 Erzwingung gegenseitiger Authentifizierung (mTLS) (H)

Für die Kommunikation zwischen hochsensiblen Datensammlern (z. B. kritische Infrastruktur-Sensoren) und der Datenbank SOLLTE eine gegenseitige Authentifizierung mittels TLS (mTLS) erzwungen werden. Sowohl der Client als auch der Server MÜSSEN ihre Identität durch Zertifikate nachweisen.

APP.4.7.A10 Verschlüsselung der gespeicherten Daten (Encryption at Rest) (H)

Werden in der Zeitreihen-Datenbank besonders schützenswerte Informationen gespeichert, SOLLTEN die Daten auf dem Speichermedium verschlüsselt werden. Dies SOLLTE durch Mechanismen des Betriebssystems oder der Datenbank selbst umgesetzt werden.

APP.4.7.A11 Begrenzung von Ressourcen für Abfragen (H)

Um Denial-of-Service-Angriffe durch missbräuchliche Abfragen zu verhindern, SOLLTEN technische Begrenzungen für Abfragen implementiert werden. Dies kann die maximale Laufzeit einer Abfrage, die Anzahl der zurückgegebenen Datenpunkte oder die maximale Kardinalität, die in einer Abfrage verarbeitet werden darf, umfassen.

APP.4.7.A12 Sicherstellung der Datenintegrität durch kryptografische Signaturen (H)

In Umgebungen mit geringem Vertrauen in das Netzwerk oder die datenliefernden Endpunkte (z. B. IoT) SOLLTEN die Messwerte vom Agenten kryptografisch signiert werden. Ein vorgeschalteter Dienst oder die Datenbank selbst SOLLTE die Signatur jeder eingehenden Nachricht überprüfen, bevor die Daten gespeichert werden, um deren Integrität zu gewährleisten.