

Sicherheitsanforderung

Hadoop

Deutsche Telekom Gruppe

Version	2.0
Datum	01.12.2025
Status	Freigegeben

Impressum

Herausgeber
Deutsche Telekom AG
Vorstandsbereich Technology & Innovation
Chief Security Officer

Reuterstrasse 65, 53315 Bonn
Deutschland

Dateiname	Dokumentennummer 3.98	Dokumententyp Sicherheitsanforderung
Version 2.0	Stand 01.12.2025	Status Freigegeben
Fachlicher Ansprechpartner Telekom Security psa.telekom.de	Gültigkeitsdauer 01.12.2025 - 30.11.2030	Freigegeben von Stefan Pütz, Leiter SEC-T-TST

Zusammenfassung

Copyright © 2025 by Deutsche Telekom AG.

Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Inhaltsverzeichnis

1.	Einleitung	4
2.	Hadoop	5
2.1.	Choice of Distribution	5
2.2.	Architecture	7
2.3.	Choice of Hadoop Software Components, Use of Plugins	9
2.4.	Encryption of Data in Rest and in Motion	10
2.5.	Data, Data Access, and Roles	11
2.6.	Basic System Configuration	19
2.7.	HDFS Configuration	27
2.8.	Kerberos Configuration	28
2.9.	Knox Configuration (Hortonworks)	30

1. Einleitung

This document addresses Hadoop installations storing and using not only pseudo- or anonymized data but also raw data. Installations of the former type require pseudo-/anonymization at the data sources or at the points of ingress to the Data Lake and according to predefined use cases. Installations of the latter type require the implementation of above-baseline measures to ensure adequate protection of sensitive data. The challenge being is to protect both large amounts of data and most sensitive personal data.

Further, within this document an on-premises installation of Hadoop is assumed, in contrast to managed or cloud installations. In the latter cases further requirements, e.g. in regard to commissioned data processing or processes in regard to interfaces to third parties, and strict audit requirements might be essential to define and comply with.

The requirements defined in this document are addressing the aspects of data privacy and data security for big data of “Design Guideline for Big Data projects based on Hadoop” by Dr. Lars Hofmann et al.

While the author of this document on security requirements strives to define distribution-independent requirements, they might be biased towards Hortonworks, as that is the distribution in place at the authors company.

2. Hadoop

2.1. Choice of Distribution

Req 1	Hadoop distributions supporting central management and central security management must be used.
-------	--

In practice, it will not be possible to ensure rollout and enforcement of consistent security policies if non-central management solutions are in place.

At the time of writing, Apache Ranger and Apache Sentry were available for central security management and Apache Ambari for central operational management.

ID: 3.98-1/2.0

Req 2	Hadoop distributions supporting authentication and authorization of all components of a Hadoop installation must be used.
-------	---

To ensure complete and consistent implementation of security policies as well as to prevent loopholes, authentication of users and services in regard to data ingest, data access, and for execution of jobs must be possible.

ID: 3.98-2/2.0

Req 3	Hadoop distributions must support the forwarding of user and service rights.
-------	--

A user or service must access data directly, e.g. data stored on a data node should not be accessed directly by a user, but through some other software component from the Hadoop set of components. In such typical case, the access rights of the original user must be used to grant data access, not the access rights of the respective other software component.

ID: 3.98-3/2.0

Req 4	Eingesetzte Software muss aus vertrauenswürdigen Quellen bezogen und auf Integrität geprüft werden.
-------	---

Die auf dem System eingesetzte Software muss aus vertrauenswürdigen Quellen bezogen und vor der Installation auf ihre Integrität geprüft werden.

Diese Anforderung bezieht sich auf sämtliche Arten von Software:

- Firmware sowie Microcode für Hardware-Komponenten
- Betriebssysteme
- Software-Bibliotheken
- Applikationen und Anwendungssoftware
- vorintegrierte Anwendungslösungen, wie z. B. Software Appliances oder Container

sowie weitere Software, die ggf. zum Einsatz kommt.

Vertrauenswürdige Quellen

Als vertrauenswürdige Bezugsquellen gelten allgemein:

- die offiziellen Vertriebs- und Bereitstellungs Kanäle des Herstellers
- dritte Distributoren, sofern sie durch den Hersteller autorisiert und legitimer Bestandteil von dessen Bereitstellungs Kanälen sind
- Internet-Downloads, sofern diese von offiziellen Bereitstellungs-Servern des Herstellers oder autorisierter Distributoren erfolgen

- (1) Bietet der Bereitstellungs-Server verschiedene Formen von Downloads an, müssen durch Verschlüsselung bzw. kryptografische Signaturen geschützte Downloads jenen ohne einen solchen Schutz vorgezogen werden.
- (2) Sichert der Bereitstellungs-Server die Transport-Schicht mittels kryptografischer Protokolle (z. B. HTTPS, sftp), müssen die zugehörig präsentierten Server-Zertifikate oder Server-Keys/Fingerprints bei jedem Download validiert werden, um die Identität des Bereitstellungs-Servers zu bestätigen; schlägt die Validierung fehl, ist der Download abubrechen und der Bereitstellungs-Server als nicht vertrauenswürdige Quelle zu bewerten.

Integritätsprüfung

Die Integritätsprüfung soll sicherstellen, dass die erhaltene Software frei von Manipulationen und Schadsoftware-Befall ist. Sofern vorhanden, müssen zur Prüfung die vom Hersteller implementierten Mechanismen verwendet werden.

Valide Mechanismen sind:

- physikalische Siegel bzw. fest aufgebrachte Echtheitszertifikate (bei Bereitstellung der Software auf physikalischen Datenträgern)
- Abgleich von kryptografischen Hash Werten (z. B. SHA256, SHA512) der erhaltenen Software gegen Soll-Werte, die der Hersteller separat benennt
- Prüfung von kryptografischen Signaturen (z. B. GPG, Zertifikate), mit denen der Hersteller seine Software versieht

Zusätzlich ist eine Überprüfung der Software mittels eines Anti-Virus- bzw. Anti-Malware Scanners empfohlen (falls der Hersteller keinen der zuvor erwähnten Integritätsschutz-Mechanismen für seine Software implementiert hat, ist diese Überprüfung verpflichtend).

Erweiterte Integritätsprüfung bei Bezug von Software aus öffentlichen Registries

Öffentliche Registries ermöglichen es Entwicklern, beliebige eigene Software-Projekte zur Nutzung bereitzustellen. Die Bandbreite umfasst sowohl Projekte von namhaften Anbietern mit kontrollierten Entwicklungsprozessen als auch von kleineren Anbietern oder Amateur-Entwicklern.

Beispiele für solche Registries sind:

- Code Registries (z. B. GitHub, Bitbucket, SourceForge, Python Package Index)
- Container Registries (z. B. Docker Hub)

Software aus öffentlichen Registries muss vor dem Einsatz einer erweiterten Integritätsprüfung unterzogen werden. Zusätzlich zu den im vorherigen Abschnitt beschriebenen Bestandteilen der Integritätsprüfung soll die erweiterte Prüfung explizit sicherstellen, dass die Software ihre Funktion tatsächlich wie beschrieben erfüllt, keine inhärenten Sicherheits-Risiken wie etwa gezielt implementierte Malware-Funktionen enthält und nicht von bekannten Sicherheits-Schwachstellen betroffen ist. Sofern die Software direkte Abhängigkeiten zu dritten Software-Projekten aufweist (sog. "Dependencies", sehr typisch in Open Source Software), die zum Einsatz der Software ebenfalls bezogen und installiert werden müssen, sind diese in die erweiterte Integritätsprüfung mit einzubeziehen.

Methoden für eine erweiterte Integritätsprüfung können beispielsweise sein:

- strikte Validierung der Projekt-/Paket-Namen (Vermeidung der Verwechslung mit maliziösen, gezielt nachgeahmten Software-Projekten)
- dynamische Code Analyse / strukturierte Funktionsprüfung in einer Test-Umgebung
- statische Code Analyse mittels eines Linters (z. B. Splint, JSLint, Pylint)
- Untersuchung mittels eines Sicherheitsschwachstellen-Scanners (z. B. Qualys, Nessus)
- Untersuchung mittels eines Container Security Scanners (z. B. JFrog Xray, Harbor, Clair, Docker Scan)
- Untersuchung mittels eines SCA (Software Composition Analysis) Tools bzw. Dependency-Scanners (z. B. OWASP Dependency Check, Snyk)

Die Prüfungsmethoden sind passend zur exakten Form der Software-Bereitstellung (Quellcode, Binaries/Artefakte, Container) auszuwählen und angemessen zu kombinieren.

Motivation: Softwarelieferketten enthalten diverse Angriffsvektoren. Ein Angreifer kann an verschiedenen Stellen an-

setzen, um Software zu manipulieren oder gezielt eigene Routinen einzubringen und die Zielumgebung, in der die Software anschließend zum Einsatz gebracht wird, zu schädigen oder unter seine Kontrolle zu bringen. Der Angriff kann auf dem Transport- bzw. Übertragungsweg geschehen oder auch auf die Bereitstellungsquelle selbst ausgeübt werden. Entsprechend erleichtert wird ein Angriff, falls Software nicht über offizielle und kontrollierte Quellen bezogen wird oder eine Integritätsprüfung ausgelassen wird.

Besonderes Risiko zeigt sich für Software, die aus öffentlichen Registries bezogen wird, da diese jedermann für die Bereitstellung von Software-Projekten offenstehen. Perfide Angriffsmethoden sind inzwischen bekannt, in denen der Angreifer zunächst eine Zeitlang sicherheitstechnisch völlig unauffällige, funktionale Software bereitstellt und sobald diese sich etabliert und eine gewisse Verbreitung gefunden hat, in zukünftigen Versionen gezielt versteckten Schadcode integriert. Andere Methoden setzen auf ähnlich klingende Projektnamen zu weit verbreiteten Bestandsprojekten oder überstimmende Versionsnummern, um manipulierte Software in darauf aufbauende Lösungen zu injizieren.

Umsetzungsbeispiel: Bezug der Software über die offiziellen Bereitstellungskanäle des Herstellers. Nach Erhalt der Software umgehend Prüfung auf Integrität mittels kryptografischer Checksummen, wie vom Hersteller bereitgestellt, sowie Freiheit von bekannter Schadsoftware mittels Anti-Malware- / Anti-Virus-Scanner. Ablage der geprüften Software auf einer internen, geschützten Dateiablage und weitere Nutzung (z. B. Rollout auf die Zielsysteme) nur noch von dort.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Modifikation von Daten
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-2/9.0

2.2. Architecture

Req 5	Hadoop-specific network segments and systems must be separated from other segments and systems by stateful packet filters and by using separate VLANs. Ingress nodes can share a dedicated network segment with high volume data sources without a separate stateful packet filter, if additional security measures are implemented.
-------	--

For Hadoop's security framework to be effective, it is crucial to use all and only components from the Hadoop set of components. To prevent circumvention of security measures, such components must be segregated from non-Hadoop components at least at the data link layer, for example by using separated and dedicated VLANs, and by using stateful packet filters for connection of different network segments.

"Separation" regarding this document implies allowing only necessary and explicitly approved traffic from machines in one segment to machines in other segments while blocking all other traffic, which in turn implies the necessity to create a maintain a communication matrix.

Due to large amounts of data potentially being ingested by specific data sources, such high volume data sources and ingress node can share a common network segment which is used solely for the purpose of transferring data from specific high volume data sources to ingress edge nodes. In such cases, host based packet filters must be used on the ingress edge nodes, e.g. iptables on Linux systems.

ID: 3.98-5/2.0

Req 6	At least four network segments must be used to separate traffic; regarding data ingest, data use and data access, monitoring and utility, and name and data nodes; separated by stateful packet filters.
-------	--

Edge nodes are the nodes through which data is ingested in the Data Lake and which control access to data stored in the Data Lake, and all systems of a Hadoop installation require management and monitoring functions. To prevent bypass of access controls those four different functions (data ingest, data use and data access, monitoring and utility, and master and data nodes from the Data Lake) require segregation from each other as well as segregation from other systems.

To note: Some Hadoop-components might not be clearly assignable to one of the four network segments, e.g. Hue, which might be used as a user access component and therefore is to be placed on an edge node or as a management component, to be placed on a monitoring and utility node.

Req 7	Edge nodes used to ingest data into the data lake must be dual homed, one leg homed in the data ingest segment and the other leg homed in the name and data node segment if access to such dual homed hosts is secured with an extra layer of security.
-------	---

This requirement attributes to the large amounts of ingested data flowing directly from edge nodes to data nodes. For performance reasons another packet filter in this flow might not be feasible. To ensure security in such cases, access to such dual homed edge nodes require extra protection, e.g. a shell control box through which all shell access is granted only through a proxy and which whitelists only allowed commands permitted to be executed on such edge nodes.

Independent from the architectural design decision, i.e. using dual homed or not dual homed edge nodes, access from edge nodes to the data lake must be possible to allow the edge nodes' HDFS clients access to the data lake.

Req 8	Non-system-management access to data in a Hadoop installation must be performed through egress edge nodes only.
-------	---

For accessing data in the Data Lake, users must connect to edge nodes, they must not access data in the data lake directly. This requirement implies that not all services available in principle in fully accessible installation can be used or that such services require special drivers being aware of edge nodes. For more detailed requirements on egress edge nodes, see Knox configuration.

Req 9	Systems used for Hadoop components must not be used for non-Hadoop related applications, i.e. Hadoop components must not share a system with application not related to Hadoop.
-------	---

To avoid bypass of security policies, resulting in risk of unauthorized data access, such sharing of systems must not take place, neither by having Hadoop components as well as non-Hadoop related applications installed under the same operating systems nor by using different virtual machines on the same platform.

Req 10	Master services must run on master nodes separated from worker services, which must run on worker nodes, separated from management services, which must run on management nodes, and user-facing services, which must run on edge nodes.
--------	--

The Hadoop architecture often distinguishes between master services/nodes and worker services/nodes, for example name nodes and data nodes. While name nodes manage access to data, data nodes actually store blocks of data. Because of the importance of master services they should not run on the same machines as worker services, as such separation allows for:

- Resource contention: E.g., user jobs exhausting resources on a worker node will not degrade essential master nodes services or even cause a denial-of-service to master node services.
- Reduction of the effects of privilege escalation: E.g., master services require different authorizations than worker nodes; in case of mixed master/worker services on one node access rights for one type of service might be used to access data of the other type of service.
- Containment of security vulnerabilities: E.g., vulnerabilities specific to worker nodes will not influence essential master node services.
- One might like to consider installing different master services on different machines for added reliability and

added protection of services, resulting again in resource contention and containment of security vulnerabilities.

Non-exhaustive list of master services:

- HDFS name node, secondary name node (standby name node), failover controller, journal node
- MapReduce job tracker and failover controller
- YARN resource manager and job history server
- Hive metastore server
- ZooKeeper server
- HBase master

Accumulo master, tracer, and garbage collector

Non-exhaustive list of worker services:

- HDFS data node
- MapReduce task tracker
- YARN node manager
- HBase region server
- Accumulo tablet server
- Solr server

Non-exhaustive list of management services:

- configuration management
- monitoring
- alerting
- software repositories
- backend databases
- DHCP servers

Non-exhaustive list of edge node services:

- HDFS HttpFS and NFS gateway
- KNOX
- client CLI tools

ID: 3.98-10/2.0

2.3. Choice of Hadoop Software Components, Use of Plugins

Req 11	Components from the Hadoop set of components must be used to cover the security framework blocks of administration (central management and consistent security), authentication (authentication of users and systems), authorization (provision access to data), audit (maintain records of data access), data protection (protection of data at rest and in motion), and perimeter security & security baseline (protection of network boundaries and systems).
--------	--

All software components must supplement and extend each other to ensure security, which is only ensured for components from the Hadoop set of components. Non-Hadoop components (i.e. components not part of the Hadoop set of components) might or might not fit, but the risk of incompatibilities and lack of long-term maintenance is quite high. Exception: As currently, i.e. at the time of writing, no Hadoop component exists covering pseudo-/anonymization, such component must be developed separately and integrated into the Hadoop installation.

ID: 3.98-11/2.0

Req 12	All non-security Hadoop components (i.e. all components allowing access to data) must use plug-ins enforcing the security policy.
--------	---

Some components can be used without enforcement of security policies. Such approach might be sensible in cases where only public, generally accessible data is used. Therefore, all non-security components must use according plug-ins, allowing for policy enforcement.

ID: 3.98-12/2.0

2.4. Encryption of Data in Rest and in Motion

Req 13	Credentials must be encrypted during transmission.
--------	--

For example, HTTPS, SSH, or SFTP must be used in favour of HTTP, Telnet, or FTP.

ID: 3.98-13/2.0

Req 14	Transmission of data between components in regard to system management or security management must be encrypted.
--------	--

All such data, e.g. data transmitted by Apache Ambari to its clients, must be protected in regard to confidentiality and integrity. Therefore, for such communications, TLS must be configured and used.

ID: 3.98-14/2.0

Req 15	Data transmitted to or accessed from the outside of the Hadoop installation must be encrypted.
--------	--

To ensure confidentiality and integrity of sensitive data, all such data must be encrypted.

ID: 3.98-15/2.0

Req 16	Administrative access to any Hadoop component must be encrypted.
--------	--

To ensure confidentiality and integrity of sensitive data, all such data must be encrypted.

ID: 3.98-16/2.0

Req 17	Data transmitted within the Hadoop installation must be encrypted if classified “confidential” or “strictly confidential”. Such encryption can be omitted in case the underlying cables, physical layer infrastructure and data link layer infrastructure is used exclusively by Hadoop systems within one Hadoop network segment and such infrastructure does not cross site boundaries.
--------	---

To ensure confidentiality and integrity of sensitive data, all such data must be encrypted.

The preferred option to do so is to use end-to-end encryption, i.e. encryption should be performed on the endpoints of the Hadoop components. Nevertheless, such data can be transmitted using encrypted site-to-site VPNs, if the VPN endpoints terminate within the Hadoop installation or the firewall separating the Hadoop installation's network segments.

The exception stated addresses cases where data is transferred from one node to another and both respective nodes are housed within the same rack, racks in the same data centre room or adjacent data centre rooms, and a private networks infrastructure with own cables and switches is installed.

ID: 3.98-17/2.0

Req 18	Data classified “confidential” or “strictly confidential” must be stored encrypted on all parts of the Hadoop installation.
--------	---

To ensure confidentiality and integrity of sensitive data, all such data must be encrypted. Compliance with this requirement will also ensure compliance with standards requiring to keep data inaccessible even if some hat physical access to a server, e.g. PCI-DSS or HIPPA.

ID: 3.98-18/2.0

2.5. Data, Data Access, and Roles

Req 19	During transformation of data from the raw data stage into cleansed data stage, all data must be labelled.
--------	--

As authorization of data access depends on role, stage, and label, all non-raw data must be labelled.

ID: 3.98-19/2.0

Req 20	Labelling of data should reflect information classification principles as defined by corporate policy.
--------	--

Each type of label must be part of one class according to corporate policy.

ID: 3.98-20/2.0

Req 21	Data not pseudo- or anonymized must be stored in the Hadoop file system separately from other data.
--------	---

Such separation can be ensured by implementing stringent staging, which stores data of different classification in different directories, combined with according policies or ACLs.

ID: 3.98-21/2.0

Req 22	The Hadoop file system must be configured to allow access only if such access is controlled via the central security management system.
--------	---

E.g., data nodes must not accept requests access to data from users directly logged in a data node. It is understood that administrative users might have more extensive access rights, possibly allowing for circumvention of the central management system.

ID: 3.98-22/2.0

Req 23	Assuming roles for production, testing, and development is mutually exclusive, e.g. one person cannot assume roles for both production and testing or for all three group of roles, within one Hadoop-installation.
--------	---

Implementation of this requirement hinders circumvention of the role concept. One person can be developer on a development installation and assuming a production role on a production system.

ID: 3.98-23/2.0

Req 24	User must not access data nodes directly, unless for system administrative purposes. Such access must take place via edge nodes only.
--------	---

For the defined Hadoop security framework being effective it is essential to strictly gate any non-administrative access through the defined gates, i.e. the edge nodes, only.

ID: 3.98-24/2.0

Req 25	Schutzbedürftige Daten müssen während der Übertragung vor nicht autorisierter Einsichtnahme und Veränderung geschützt werden.
--------	---

Der Schutzbedarf von zu übertragenden Daten ist abhängig von deren Einstufung (beispielsweise nach geltenden gesetzlichen Datenschutzvorgaben, regulatorischen Auflagen, vertraglichen Verpflichtungen), dem Schadenspotential bei deren Missbrauch, sowie weiteren relevanten Faktoren (beispielsweise der Übertragung über öffentliche Netze). Art und Umfang der Schutzmaßnahmen müssen entsprechend angemessen gewählt werden.

Authentisierungsmerkmale wie beispielsweise Passwörter oder Tokens gelten grundsätzlich als schutzbedürftige Daten. Ebenfalls grundsätzlich als schutzbedürftig gelten Daten, die Funktionalität und sicherheitsrelevantes Verhalten eines Systems bestimmen (z. B. Updates & Patches, Konfigurationsparameter, Fernwartungen, Steuerung über APIs).

Die Einhaltung der Schutzziele Vertraulichkeit und Integrität muss während der Übertragung von schutzbedürftigen Daten durchgängig gewährleistet werden.

I.d.R. erfordert dies eine Implementation von kryptografischen Verfahren (z. B. Verschlüsselung, Signaturen, Hashes). Kryptografische Verfahren können

- vor der Übertragung direkt auf die Daten angewendet werden, was eine anschließende Übertragung der Daten auch über unsichere Kanäle vertretbar machen kann
- auf dem Übertragungs-Kanal eingesetzt sein, um einen sicheren Kanal zu schaffen und beliebige Daten, die durch diesen geleitet werden, zu schützen
- oder auch als Kombination aus beidem umgesetzt sein.

Bei der Übertragung von Daten eingesetzte kryptografische Verfahren müssen für diesen Zweck anerkannt wirksam sein und dürfen keine bekannten Schwachstellen aufweisen.

Motivation: Die Übertragung von Daten ohne einen adäquaten Schutz ermöglicht es einem Angreifer, diese unberechtigt abzufangen, zu nutzen, zu verbreiten, zu verändern oder aus der Übertragung zu entfernen. Dies eröffnet potenziell weitere Angriffsvektoren auf die unmittelbaren Zielsysteme sowie angeschlossene weitere Systeme und kann zu erheblichen Ausfällen, Kontrollverlusten und Schäden sowie resultierenden Strafzahlungsforderungen und Reputationsverlusten gegenüber Kunden und Geschäftspartnern führen.

Umsetzungsbeispiel: [Beispiel 1]

Vertrauliche Dokumente werden vor dem Versand per E-Mail an den Kunden verschlüsselt.

[Beispiel 2]

Ein Administrator konfiguriert eine neue Cloud-Anwendung über das Internet. Der Zugriff erfolgt über eine TLS-verschlüsselte Verbindung ("https").

[Beispiel 3]

Ein System bezieht automatische Software-Updates von einem Update-Server. Der Update-Server liefert die Software Updates kryptografisch signiert aus. Das System kann die empfangenen Software-Updates somit validieren und sicher ausschließen, dass diese während der Übertragung manipuliert wurden.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Störung der Verfügbarkeit
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-15/9.0

Req 26	Gespeicherte schutzbedürftige Daten müssen vor nicht autorisierter Einsichtnahme, Veränderung und Löschung geschützt werden.
--------	--

Der Schutzbedarf für gespeicherte Daten ist abhängig von deren Einstufung (beispielsweise nach geltenden gesetzlichen Datenschutzvorgaben, regulatorischen Auflagen, vertraglichen Verpflichtungen), dem Schadenspotential bei deren Missbrauch, sowie weiteren relevanten Faktoren (beispielsweise dem Ort der Speicherung). Art und Umfang von Schutzmaßnahmen müssen entsprechend angemessen gewählt werden.

Gespeicherte Authentisierungsmerkmale wie beispielsweise Passwörter, Private Keys, Tokens oder Zertifikate gelten grundsätzlich als schutzbedürftig. Ebenfalls als grundsätzlich schutzbedürftig gelten Daten, die Funktionalität und sicherheitsrelevantes Verhalten eines Systems bestimmen (z. B. System-Konfigurationsdateien, Betriebssysteme und Kernel, Treiber).

Die Einhaltung der Schutzziele Vertraulichkeit, Integrität und Verfügbarkeit muss für gespeicherte schutzbedürftige Daten durchgängig gewährleistet werden. Dies gilt auch während einer lediglich kurzfristigen Speicherung (z. B. bei der Zwischenspeicherung in einem Web Cache oder in einem temporären Ordner innerhalb einer Datenverarbeitungskette).

Grundlegend muss der Zugriff auf schutzbedürftige Daten in einem System auf Basis von technisch implementierten Berechtigungsvergaben und -kontrollen vollumfassend reglementiert werden.

Reicht eine solche technische Zugriffskontrolle allein nicht mehr aus, um den erforderlichen Schutzbedarf gespeicherter Daten zu gewährleisten, oder kann deren Wirksamkeit nicht durchgängig sichergestellt werden, müssen zusätzlich kryptografische Verfahren (z. B. Verschlüsselung, Signierung, Hashing) implementiert werden. Bei der Speicherung von Daten eingesetzte kryptografische Verfahren müssen für diesen Zweck anerkannt wirksam sein und dürfen keine bekannten Schwachstellen aufweisen.

Motivation: Die Speicherung von Daten auf einem System ohne einen adäquaten Schutz ermöglicht es einem Angreifer, diese unberechtigt einzusehen, zu nutzen, zu verbreiten, zu verändern, oder zu vernichten. Dies eröffnet potenziell zusätzliche Angriffsvektoren auf die unmittelbaren sowie angeschlossene weitere Systeme und kann zu erheblichen Ausfällen, Kontrollverlusten und Schäden sowie resultierenden Strafzahlungsforderungen und Reputationsverlusten gegenüber Kunden und Geschäftspartnern führen.

Umsetzungsbeispiel: [Beispiel 1]

Ein System exportiert Daten für einen Transport auf einen mobilen Datenträger. Da die technische Zugriffskontrolle des Systems auf Dateiberechtigungsebene nicht mehr greift, sobald der mobile Datenträger aus dem System entfernt wird, müssen zusätzliche Maßnahmen zum Schutz der Daten getroffen sein. Bevor das System die Daten auf den mobilen Datenträger schreibt, werden sie entsprechend mit einem passenden Verfahren verschlüsselt. Der zugehörige Schlüssel wird auf einem separaten Kanal ausgetauscht, so dass die Daten im legitimen Zielsystem wieder entschlüsselt und verarbeitet werden können. Ein Angreifer, der den mobilen Datenträger in seinen Besitz bringt, hat hingegen keinen Zugriff auf die Daten.

[Beispiel 2]

In der lokalen Benutzer-Datenbank eines Systems werden nur kryptografische Hashes der Passwörter gespeichert, die mit einem als sicher geltenden Password-Hashing Verfahren erzeugt wurden. Für das System sind diese Hashes ausreichend, um die Benutzer bei einer Anmeldung am System zu authentifizieren. Sollte ein Angreifer die Benutzer-Datenbank kopieren können, gelangt er somit jedoch nicht unmittelbar in den Besitz von Klartext-Passwörtern, mit denen er sich am System im Namen der Benutzer anmelden könnte.

[Beispiel 3]

Auf einem System sind die Konfigurationsdateien des Web-Servers ausschließlich für den legitimen Admin schreibbar, in dem im Dateisystem entsprechende Berechtigungen gesetzt wurden. Die Zugriffskontrolle des Betriebssystem-Kernels verweigert somit allen anderen Benutzern des Systems, Veränderungen an den Konfigurationsdateien des Web-Servers vorzunehmen; einschließlich dem Web-Server Dienste-Konto selbst, was auch die Angriffsfläche von außen bei Schwachstellen im Web-Server verkleinert.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Störung der Verfügbarkeit
- Unbemerkt durchführbare Angriffe
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

Req 27	Die Nutzung von schutzbedürftigen Funktionen des Systems sowie der Zugriff auf interne oder vertrauliche Daten darf nicht ohne vorhergehende Authentifizierung und Autorisierung ermöglicht werden.
--------	---

Die Nutzung von schutzbedürftigen Funktionen des Systems sowie der Zugriff auf Daten, die als intern oder vertraulich eingestuft sind, darf erst nach einer eindeutigen Identifizierung und erfolgreichen Authentifizierung des Benutzers anhand eines Benutzerkontos ermöglicht werden. Ein Benutzerkonto besteht aus einem eindeutigen Benutzernamen und zugehörigen Authentisierungsmerkmalen. Zudem muss verifiziert werden, dass der Benutzer innerhalb der ihm im System zugeordneten Benutzerrolle für den Zugriff auf die betroffenen Funktionen und Daten legitimiert ist.

Eine Ausnahme hiervon sind Funktionen und Daten, die uneingeschränkt öffentlich genutzt werden dürfen; beispielsweise der Bereich einer Web-Seite im Internet, in dem nur öffentliche Informationen bereitgestellt werden.

Beispiele für Funktionen, die eine vorhergehende Authentifizierung erfordern:

- Fernzugriff auf Netzdienste (wie SSH, SFTP, Web-Dienste)
- lokaler Zugriff auf die Management-Konsole
- lokale Nutzung von Betriebssystem und Anwendungen

Diese Anforderung gilt ohne Einschränkung auch für jegliche maschinellen Zugriffe auf das System (Hier erfolgt die Umsetzung in der Regel durch den Einsatz von sogenannten "technischen Benutzerkonten", auch "Machine-2-Machine" oder "M2M"-Konten genannt).

Motivation: Die eindeutige Authentifizierung und Autorisierung von Zugriffen auf ein System sind elementar, um Funktionen und Daten vor Missbrauch zu schützen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Abstreiten von durchgeführten Aktionen

Req 28	Benutzerkonten müssen mit mindestens zwei Authentisierungsmerkmalen aus unterschiedlichen Faktoren geschützt werden.
--------	--

Sämtliche Benutzerkonten des Systems müssen vor einer unberechtigten Nutzung geschützt werden. Dies ist durch Verwendung von Authentisierungsmerkmalen umzusetzen, die eine eindeutige Identifizierung des legitimen Benutzers ermöglichen.

Authentisierungsmerkmale werden in übergeordneten Gruppen, sogenannten "Faktoren", differenziert:

- Faktor WISSEN: "etwas, das nur der legitime Benutzer weiss"
- Faktor BESITZ: "etwas, das nur der legitime Benutzer besitzt"
- Faktor INHÄRENZ: "etwas, das nur der legitime Benutzer ist"

Es müssen mindestens zwei Authentisierungsmerkmale miteinander kombiniert werden.

Diese müssen aus unterschiedlichen Faktoren stammen ("Multi-Faktor-Authentisierung"; kurz: "MFA").

Eine Kombination von Authentisierungsmerkmalen desselben Faktors (Beispiel: zwei unterschiedliche PINs) ist nicht zulässig.

Beispiele für zulässige Authentisierungsmerkmale

- aus dem Faktor WISSEN: PINs, Passphrases (Passwörter*)
- aus dem Faktor BESITZ: Smartcards, FIDO2 Level 2 Security Keys, Device-bound Passkeys, in einem TPM/HSM gespeicherte kryptografische Schlüssel, OTPs (One-Time Passwords) generiert in einer zum Einsatz in der Deutschen Telekom freigegebenen Authenticator App
- aus dem Faktor INHÄRENZ: biometrische Merkmale wie Fingerabdruck, Iris- bzw. Retina-Scans, Gesichtserkennung

***) Wichtiger Hinweis zum Einsatz von Passwörtern**

Passwörter müssen als Authentisierungsmerkmal strikt vermieden werden.

Sie sind nur dann zulässig, falls sicherere Alternativen nicht umgesetzt werden können.

Grundsätzlich nicht zulässig sind Authentisierungsmerkmale, die leicht gefälscht oder manipuliert werden können.

Dazu gehören:

- Rufnummern und Anrufer-IDs
- IP-Adressen
- geografische Standorte
- VPN-Zugehörigkeiten und Netzwerkinformationen
- Status von Clients, etwa von Arbeitsplatzsystemen oder mobilen Endgeräten

Hinweis zu technischen Benutzerkonten

Für den maschinellen Zugriff eingesetzte technische Benutzerkonten (auch unter dem Begriff "Machine-to-Machine" bzw. "M2M"-Konten bekannt) sind von der Verpflichtung zur Multi-Faktor-Authentisierung ausgenommen, da eine solche dort i.d.R. nicht umsetzbar ist.

Motivation: Benutzerkonten, die nicht über angemessene Authentisierungsmerkmale geschützt sind, können von einem Angreifer missbraucht werden, um unberechtigt auf ein System und die darauf gespeicherten Daten und Anwendungen zuzugreifen. Aufgrund vielfältiger gängiger Angriffsmethoden ist der Schutz von Benutzerkonten durch ein einzelnes Authentisierungsmerkmal mittlerweile als schwach zu bewerten, insbesondere, falls hier das klassische Passwort zum Einsatz kommt. Eine Kombination von Authentisierungsmerkmalen aus unterschiedlichen Faktoren erhöht die Sicherheit für Benutzerkonten signifikant.

Umsetzungsbeispiel: Beispiele für MFA

MFA Kombination	Beschreibung	Anmerkung zur Anwendung
Telekom "myCard" plus PIN-Eingabe	1. Faktor: in Hardware implementiertes Zertifikat ("Besitz") 2. Faktor: PIN ("Wissen")	(keine)
FIDO2 Level 2 Security Key plus Fingerabdruck	1. Faktor: in Hardware gespeicherter Crypto-Schlüssel ("Besitz") 2. Faktor: Biometrisches Merkmal ("Inhärenz")	Die Biometrie-Funktion muss vorab konfiguriert werden, da der Biometrie-Sensor sonst nur als "Taste" funktioniert und somit der 2. Faktor ("Inhärenz") nicht gegeben wäre.
Device-bound Passkey in WH4B* auf Nucleus PC plus PIN	1. Faktor: in Hardware gespeicherter Crypto-Schlüssel ("Besitz") 2. Faktor: PIN ("Wissen")	(keine)

Device-bound Passkey in Secure Enclave auf iOS Device plus FaceID	1. Faktor: in Hardware gespeicherter Crypto-Schlüssel ("Besitz") 2. Faktor: Biometrisches Merkmal ("Inhärenz")	(keine)
Passwort plus freigegebene OTP App (z. B. MS Authenticator)	1. Faktor: Passwort ("Wissen") 2. Faktor: spezifisch generiertes One Time Password ("Besitz")	Passwörter sind nur dann zulässig, falls sicherere Alternativen nicht umgesetzt werden können. Die Synchronisation der Authenticator App über einen Clouddienst sowie das Teilen (z. B. über einen Passwortmanager) des OTP ist nicht zulässig.

*) WH4B: "Windows Hello for Business"

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Abstreiten von durchgeführten Aktionen

ID: 3.01-20/9.0

Req 29	Privilegierte Benutzerkonten müssen mit mindestens zwei Authentisierungsmerkmalen geschützt werden, die aus unterschiedlichen Faktoren stammen und erhöhte Resilienz gegen Phishing-Angriffe bieten.
--------	--

Ein privilegiertes Benutzerkonto ist ein Benutzerkonto, welches über erweiterte Berechtigungen innerhalb eines Systems verfügt. Erweiterte Berechtigungen ermöglichen einen Zugriff auf Konfigurationseinstellungen, Funktionen oder Datenbestände, der regulären Benutzern des Systems nicht zu Verfügung steht. In direkter Abhängigkeit zu den besonderen Aufgaben, die mittels eines privilegierten Benutzerkontos innerhalb eines Systems wahrgenommen werden, können die zugewiesenen erweiterten Berechtigungen spezifisch limitiert sein oder einen vollständig uneingeschränkten Systemzugriff umfassen.

Beispiele für privilegierte Benutzerkonten:

- Konten für Administrations-, Wartungs- oder Entstörungsaufgaben
- Konten für Benutzer-Verwaltungsaufgaben (z. B. Anlegen/Löschen von Benutzern; Zuweisen von Berechtigungen; Passwort-Rücksetzungen)
- Konten, die zur Freigabe, Auslösung oder Unterbindung von geschäftskritischen Vorgängen berechtigt sind
- Konten mit Zugriff auf Daten, die im Interesse des DTAG-Konzerns, dessen Kunden oder der Öffentlichkeit als SCD (Sensitive Customer Data) eingestuft sind
- Konten, die in hohem Umfang auf lt. EU-DSGVO als „personenbezogen“ definierte Daten zugreifen können (z. B. Massenabruf von größeren Teilen oder des vollständigen Datenbestands)

Zum Schutz privilegierter Konten müssen mindestens zwei Authentisierungsmerkmale miteinander kombiniert werden.

Diese müssen aus unterschiedlichen Faktoren ("WISSEN", "BESITZ" oder "INHÄRENZ") stammen.

Eine Kombination von Authentisierungsmerkmalen desselben Faktors ist nicht zulässig.

Mindestens eines der Authentisierungsmerkmale darf nicht anfällig gegenüber Phishing-Angriffen sein.

Bei einem Phishing-Angriff wird der legitime Nutzer trickreich manipuliert, um seine Anmeldedaten dem Angreifer of-

fenzulegen, beispielsweise mittels gefälschter Anmelde-Seiten. Ein Phishing-Angriff kann auch gegen eine Kombination aus mehreren Authentisierungsmerkmalen erfolgen.

Phishing-resiliente Authentisierungsmerkmale verhindern Angriffe anhand ihrer technischen Umsetzung. Diese Authentisierungsmerkmale nutzen kryptographische Verfahren und stellen eine starke Bindung zwischen dem Nutzer, dem Gerät und dem Dienst her. Sie erfordern während der Authentisierung zusätzlich eine manuelle Nutzerinteraktion, die nicht automatisiert werden kann; beispielsweise, einen Fingerabdruck-Scan an einem Hardware Token abzugeben.

Als nicht anfällig gegenüber Phishing-Angriffen gelten

- FIDO2 Level 2 Security Keys
- in Hardware implementierte Zertifikate

Wichtiger Hinweis zum Einsatz von Passwörtern

Passwörter müssen als Authentisierungsmerkmal strikt vermieden werden.

Sie sind nur dann zulässig, falls sicherere Alternativen nicht umgesetzt werden können.

Hinweis zu technischen Benutzerkonten

Für den maschinellen Zugriff eingesetzte technische Benutzerkonten (auch unter dem Begriff "Machine-to-Machine" bzw. "M2M"-Konten bekannt) sind von der Verpflichtung zur Multi-Faktor-Authentisierung ausgenommen, da eine solche dort i.d.R. nicht umsetzbar ist.

Motivation: Privilegierte Benutzerkonten stellen ein erhöhtes Risiko für die Sicherheit eines Systems dar. Kompromittiert ein Angreifer erfolgreich ein solches Benutzerkonto, erhält er weitreichende Berechtigungen, mit denen er das System oder Systemanteile unter seine Kontrolle bringen, die Systemfunktionen stören, verarbeitete Daten einsehen/manipulieren oder geschäftskritische Vorgänge beeinflussen kann. Die Kombination von mehreren Authentisierungsmerkmalen unterschiedlicher Art minimiert das Risiko der Kompromittierung eines Benutzerkontos bereits signifikant, jedoch muss für privilegierte Benutzerkonten in besonderem Maße auf Resilienz gegen Phishing-Angriffe geachtet werden. Die zu beobachtenden Phishing-Angriffe werden zunehmend trickreicher und für Nutzer schwerer zu erkennen, so dass eine Multifaktor-Authentifizierung ohne erhöhte Resilienz gegen Phishing nicht mehr als ausreichend für privilegierte Benutzerkonten zu bewerten ist.

Umsetzungsbeispiel: Beispiele für MFA mit erhöhter Resilienz gegen Phishing-Angriffe

MFA Kombination	Beschreibung	Anmerkung zur Anwendung
Telekom "myCard" plus PIN-Eingabe	1. Faktor: in Hardware implementiertes Zertifikat ("Besitz") 2. Faktor: PIN ("Wissen")	(keine)
FIDO2 Level 2 Security Key plus Fingerabdruck	1. Faktor: in Hardware gespeicherter Crypto-Schlüssel ("Besitz") 2. Faktor: Biometrisches Merkmal ("Inhärenz")	Die Biometrie-Funktion muss vorab konfiguriert werden, da der Biometrie-Sensor sonst nur als "Taste" funktioniert und somit der 2. Faktor ("Inhärenz") nicht gegeben wäre.
FIDO2 Level 2 Security Key plus PIN-Eingabe	1. Faktor: in Hardware gespeicherter Crypto-Schlüssel ("Besitz") 2. Faktor: PIN ("Wissen")	(keine)

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Abstreiten von durchgeführten Aktionen

Req 30 Benutzerkonten müssen die eindeutige Identifizierung des Benutzers sicherstellen.

Benutzer müssen eindeutig von einem System identifiziert werden können.

Dies wird typischerweise dadurch erreicht, dass für jeden Benutzer ein individuelles Benutzerkonto genutzt wird.

Sogenannte Gruppenkonten, die sich dadurch auszeichnen, dass sie von mehreren Personen gemeinsam genutzt werden, dürfen nicht verwendet werden. Dies gilt uneingeschränkt auch für privilegierte Benutzerkonten. Die meisten Systeme verfügen nach der Grundinstallation zunächst nur über ein einzelnes Benutzerkonto mit administrativen Berechtigungen. Soll das System von mehreren Personen administriert werden, muss jede dieser Personen ein persönliches, individuelles Benutzerkonto verwenden, dem entsprechende administrative Berechtigungen bzw. Rollen zugewiesen sind.

Eine Besonderheit stellen sogenannte technische Benutzerkonten dar. Diese werden für Authentifizierung und Autorisierung von Systemen untereinander oder von Anwendungen auf einem System genutzt und können damit nicht einer konkreten Person zugewiesen werden. Solche Benutzerkonten müssen individuell pro System oder pro Anwendung vergeben werden. Hierbei ist allerdings sicherzustellen, dass eine missbräuchliche Nutzung solcher Benutzerkonten nicht möglich ist.

Möglichkeiten, um eine missbräuchliche Nutzung solcher Benutzerkonten durch Personen zu verhindern, sind:

- Konfiguration eines den Sicherheitsanforderungen entsprechenden Passwortes, dass nur einem kleinstmöglichen Kreis von Administratoren bekannt ist.
- Konfiguration des entsprechenden Benutzerkontos, so dass dieses nur lokal verwendet werden kann und eine interaktive Anmeldung nicht möglich ist.
- Verwendung eines Verfahrens mit öffentlichem und privatem Schlüssel oder Zertifikaten für die Authentifizierung des entsprechenden Benutzerkontos.
- Einschränkung des Zugriffs über ein Netz auf legitimierte Systeme.

Weitere Lösungen sind im Einzelfall auf ihre Verwendbarkeit zu prüfen.

Motivation: Die eindeutige Identifizierung eines Benutzers ist Voraussetzung, um einem Benutzer Berechtigungen zuweisen zu können, die für die Durchführung seiner Aufgaben auf dem System notwendig sind. Nur so kann eine adäquate Zugriffskontrolle auf Daten und Dienste eines Systems erreicht und der missbräuchliche Zugriff verhindert werden. Des Weiteren wird hierdurch ermöglicht, dass Tätigkeiten und Aktionen auf einem System protokolliert und einem Benutzer zugeordnet werden können.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Abstreiten von durchgeführten Aktionen

Req 31 Die Berechtigungen von Benutzern und Anwendungen müssen auf den zur Erfüllung ihrer Aufgaben benötigten Umfang beschränkt werden.

Die Berechtigungen auf einem System müssen so weit eingeschränkt werden, dass ein Benutzer nur auf Daten zugreifen und Funktionen nutzen kann, die er im Rahmen seiner Arbeit benötigt. Entsprechende Berechtigungen sind auch für den Zugriff auf Dateien zu vergeben, die Bestandteil des Betriebssystems oder von Anwendungen sind oder von diesen erzeugt werden (z. B. Konfigurations- und Protokollierungsdateien).

Neben dem Zugriff auf Daten muss auch die Ausführung von Anwendungen und deren Bestandteilen mit möglichst niedrigen Berechtigungen erfolgen. Anwendungen sollten nicht mit Administrator- oder Systemberechtigungen aus-

geführt werden.

Motivation: Werden einem Benutzer zu weitreichende Berechtigungen auf einem System gewährt, kann er in einem Umfang auf Daten und Anwendungen zugreifen, der zur Erfüllung der zugewiesenen Aufgaben nicht erforderlich ist. Dies erzeugt ein unnötig erhöhtes Risiko im Falle eines Missbrauchs, insbesondere auch, falls der Benutzer oder dessen Benutzerkonto durch einen Angreifer kompromittiert werden.

Anwendungen, die über zu weitreichende Berechtigungen verfügen, können von einem Angreifer missbraucht werden, um unbefugte Zugänge zu erhalten oder zu erweitern und damit einen Zugriff auf schutzbedürftige Daten und Systembereiche zu erlangen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Abstreiten von durchgeführten Aktionen

ID: 3.01-23/9.0

2.6. Basic System Configuration

Req 32	Hadoop installation must be configured to run in “secure mode”.
--------	---

By default Hadoop runs in non-secure mode in which no actual authentication is required. By configuring Hadoop to run in secure mode, each user and service needs to authenticate using Kerberos in order to use Hadoop services.

ID: 3.98-32/2.0

Req 33	Big Data installations is the ability to quickly add and remove nodes. Assignment of configuration data like IP addresses must be centrally controlled e.g. by DHCP.
--------	--

Key advantage of Big Data installations is the ability to quickly add and remove nodes to adapt to current processing and storage requirements. To do so in a controlled and secure way and to ensure consistent and central system management, manual assignment of configuration data like IP addresses is not feasible. Therefore, DHCP should be used for data nodes.

ID: 3.98-33/2.0

Req 34	For remote administration of nodes of a Hadoop installation Secure Shell (SSH) must be used via Admin-LAN, but no other protocols. Exception: In case the Hadoop distribution used provides integrated means for administration to ensure central administration, those means can be used, even if not SSH, as long as they use encrypted protocols.
--------	--

To ensure effective security of Big Data installations central administration and management is essential. Therefore, other protocols than SSH are acceptable, as long as the protocols used transmit user data encrypted.

ID: 3.98-34/2.0

Req 35	Each Hadoop component, i.e. each daemon, must run as a different Unix user.
--------	---

HDFS-, YARN-, MapReduce JobHistory- and other daemons must all run as different Unix users.

ID: 3.98-35/2.0

Req 36 Unix users of Hadoop components, i.e. each daemon, must not be the root user.

As root is the privileged system user it must not be used for Hadoop components as vulnerabilities in a component can result in much greater damage and in circumvention of authorization policies if the root user is used.

ID: 3.98-36/2.0

Req 37 Hadoop components, i.e. each daemon, must share a common Unix group.

Recommended is the use of "hadoop" for the name of such group.

ID: 3.98-37/2.0

Req 38 All hosts of a Hadoop installation must be time-synchronized with trusted timeservers.

As Kerberos is essential for Hadoop security and as Kerberos is time-sensitive, all hosts must be time-synchronized with trusted timeservers. Such trust will usually be accomplished only by using timeservers controlled by a Deutsche Telekom group company. If clocks differ by five or more minutes, authentication will fail.

ID: 3.98-38/2.0

Req 39 Certificates used to secure access to components from outside a production Hadoop installation must be issued by a standard certification authority.

While the use of self-signed certificates or other certificates not signed by a recognized certification authority is technically possible, for production systems only certificates issued by standard certification authorities must be used, if used for securing access from outside of the respective Hadoop installation.
As such standard certification authority a globally operating one should be used, e.g. TeleSec of Deutsche Telekom.

ID: 3.98-39/2.0

Req 40 Authentication of users of management tools must be based on existing central directory services.

In most cases, such central directory service will be an existing Active Directory server used for authentication of users accessing office systems like workstations or of users accessing production systems.
It is understood that administrators must login as local administrators on some systems for some tasks. Nevertheless, login on a management system using existing central directory services must occur before.

ID: 3.98-40/2.0

Req 41 The central security management of Hadoop (e.g. Apache Ranger, used by Hortonworks) must synchronise its user base via TLS.

Typically, many users will directly or indirectly have access to parts of the Data Lake. Therefore, implementing and maintaining another user directory in addition to the one used for authentication to use office infrastructure components will not be feasible; the solution being is to synchronise with an existing directory.

ID: 3.98-41/2.0

Req 42	The central security management of Hadoop should synchronise its user base continually; such synchronisation must take place at the very least once per day. Synchronisation of administrative user can be performed manually as required.
--------	--

Such synchronisation is necessary to have changes in the user base, e.g. by termination of work contracts and new hires.

ID: 3.98-42/2.0

Req 43	Auditing must be enabled.
--------	---------------------------

To allow for investigation of misuse and to detect loopholes in the use concept, enabling of auditing is required. Auditing must be enabled at least on the central security management component (which is Apache Ranger on Hortonworks installations).

ID: 3.98-43/2.0

Req 44	Sicherheitsrelevante Protokollierungsdaten müssen echtzeitnah an eine Erkennungs- und Reaktionslösung für Sicherheitsvorfälle bereitgestellt werden.
--------	--

Eine zulässige Erkennungs- und Reaktionslösung im Sinne dieser Anforderung muss die folgenden Kriterien erfüllen:

- Es erfolgt eine kontinuierliche Analyse bzw. Korrelation der aus angebundenen IT/NT-Systemen bereitgestellten Protokollierungsdaten.
- Die Identifizierung bedrohlicher Aktivitäten und laufender Angriffe findet auf Basis einer fortlaufend aktuell gehaltenen Erkennungsmuster-Datenbank statt.
- Integrierte Alarmierungsprozesse lösen im Fall eines Fundes die Einleitung von Gegenmaßnahmen aus.
- Sicherheitsexperten sind aktiv involviert in den Betrieb der Lösung sowie die Auswertung von Alarmen.

Eine solche Lösung ist in ihrer Funktion nicht mit einem Log Server gleichzusetzen. Entsprechend muss die Bereitstellung an eine Erkennungs- und Reaktionslösung stets zusätzlich zu einer vorhandenen Weiterleitung an einen Log Server erfolgen. Eine Ausnahme besteht, falls der genutzte Log Server seinerseits die Protokollierungsdaten umgehend nach Erhalt zusätzlich an eine Erkennungs- und Reaktionslösung bereitstellt. In diesem Fall ist es ausreichend, dass das System die Protokollierungsdaten nur an den Log Server weiterleitet.

Die Protokollierungsdaten müssen ab dem Zeitpunkt ihrer Erzeugung im System innerhalb von 5 Minuten an die Erkennungs- und Reaktionslösung bereitgestellt werden ("echtzeitnah").

Die konkret zu unterstützenden technischen Schnittstellen und Protokolle für die Bereitstellung werden durch die im Einsatzumfeld des Systems verfügbare Erkennungs- und Reaktionslösung sowie die technische Basis des Systems bestimmt.

Motivation: IT/NT-Systeme sind permanent sicherheitstechnischen Bedrohungen und Angriffen ausgesetzt. Mittels einer zentralen Analyse und der Korrelation erfasster System-Ereignisse ganzer IT-Landschaften in nahezu Echtzeit können aktuelle Bedrohungen und laufende Angriffe auch über Systemgrenzen hinaus erkannt und zeitnah passende Gegenmaßnahmen eingeleitet werden. Die unverzügerte Bereitstellung von Protokollierungsdaten an die Erkennungs- und Reaktionslösung ist essenziell für deren Effektivität.

Umsetzungsbeispiel: **Beispiel 1:** Ein System auf Basis eines klassischen Servers erfasst sicherheitsrelevante Protokollierungsdaten im lokalen Syslog oder Windows Event Log des Betriebssystems. Die Protokollierungsdaten werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and Event Management) weitergeleitet.

Beispiel 2: Auf einem System wird ein EDR ("Endpoint Detection and Response") Agent ausgeführt. Der EDR-Agent leitet die von ihm auf dem System erfassten sicherheitsrelevanten Protokollierungsdaten kontinuierlich an das zentrale Backend der EDR-Lösung aus, auf dem die weitere Analyse, Angriffserkennung und Alarmbehandlung erfolgt.

Beispiel 3: Netzwerk-Komponenten stellen Ereignisse an ein SIEM bereit, die es ermöglichen, verdächtiges Kommunikationsverhalten wie Netzwerk-Scans, unzulässige Datenflüsse zwischen Netz-Segmenten oder aktive Verbindungen bzw. Verbindungsversuche zu verdächtigen Zielen im Internet zu entdecken. Im speziellen liefern Gateway Router aus dem CNDTAG "Netflow" Logs, während Forward Proxies Zugriffe auf Domänen und/oder IP-Adressen melden und Firewalls blockierte Verbindungsversuche loggen.

Beispiel 4: Auf Basis von Cloud-nativen Funktionen entwickelte Systeme erzeugen Protokollierungsdaten mittels der Cloud-eigenen Protokollierungs- und Überwachungsinfrastruktur direkt aus diesen Funktionen heraus. In einem solchen System wird die Erzeugung sicherheitsrelevanter Protokollierungsdaten in den entsprechenden Konfigurationsoptionen der eingesetzten Cloud-nativen Funktionen gezielt aktiviert. Die entstehenden Protokollierungsdaten werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and Event Management) übertragen. Einige Cloud-Anbieter leisten eine gewisse Angriffsmustererkennung bereits in der Cloud-eigenen Überwachungsinfrastruktur und erzeugen daraus spezifische Cloud-Alarme. Diese werden zusätzlich an das SIEM übertragen.

Beispiel 5: Ein System basiert auf einer SaaS ("Software as a Service") Applikation eines Cloud-Anbieters. Typischerweise verantwortet im SaaS Modell der Cloud-Anbieter alle operativen Aspekte der Applikation, inkl. des Sicherheitsmanagements. Der Cloud-Kunde selbst hat keinen Einfluss auf die zugrunde liegende technische Infrastruktur oder den Betrieb der Applikation. Anhand von Vertragsvereinbarungen, anerkannten Zertifikaten und dokumentierten Selbstauskünften des Cloud-Anbieters wird formell überprüft, dass der Cloud-Anbieter eine Erkennungs- und Reaktionslösung für Sicherheitsvorfälle implementiert hat. Einige Cloud-Anbieter erzeugen zusätzlich dedizierte Protokollierungsdaten über sicherheitsrelevante Aktivitäten innerhalb der separaten Kunden-Mandanten ihrer SaaS Applikation (beispielsweise zu Änderungen von Konfigurationseinstellungen, Berechtigungsvergaben oder Benutzer-Konten innerhalb der Mandanten) und stellen diese den Cloud-Kunden zu Verfügung. Diese sogenannten Audit-Logs werden kontinuierlich zur weiteren Analyse, Angriffserkennung und Alarmbehandlung an ein SIEM (Security Information and Event Management) übertragen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unbemerkt durchführbare Angriffe

ID: 3.01-37/9.0

Req 45	Nicht benötigte (Default-) Datenbanken auf dem Datenbanksystem müssen gelöscht werden.
--------	--

Motivation: Bei der Installation von Datenbanksystemen werden oftmals Test- oder Übungsdatenbanken installiert, die im Wirkbetrieb der Datenbank nicht mehr erforderlich sind. In der Vergangenheit zeigten diese Testdatenbanken Schwachstellen, die es einem Angreifer ermöglichen, privilegierte Rechte auf dem Datenbanksystem zu erlangen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-7/9.0

Req 46	Ein Datenbankdienst darf nicht mit Root-Rechten oder anderen administrativen Rechten des Betriebssystems laufen.
--------	--

Motivation: Läuft ein Datenbankdienst mit hochprivilegierten Betriebssystemrechten (Root, Administrator), kann ein Angreifer durch die Ausnutzung einer Schwachstelle ein Datenbanksystem vollständig übernehmen (Zugriff auf das Filesystem bzw. Ausführung von Dateien).

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-18/9.0

Req 47 Nicht benötigte (Default-)User und (Default-)Rollen müssen gelöscht werden.

Motivation: Bei der Installation eines Datenbanksystems werden eine Vielzahl von Benutzern und Rollen automatisch installiert, die für den Betrieb der Datenbank nicht benötigt werden. Diese sind z. B. für Übungs- und Test-Datenbanken gedacht. Ein Angreifer kann durch die Kenntnis von Default-Usern und Rollen (privilegierten) Zugriff auf das Datenbanksystem erlangen. Daher sind Benutzer und Rollen, die nicht für die Funktionen der Datenbank gebraucht werden, zu löschen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System

ID: 3.16-8/9.0

Req 48 In Ausgaben und Meldungen dürfen keine Informationen über interne Strukturen des Systems preisgegeben werden.

Informationen über die internen Strukturen eines Systems einschließlich der dort verwendeten Komponenten und entsprechende Implementationsdetails gelten grundsätzlich als schutzbedürftig.

Im Allgemeinen betrifft dies Informationen über

- Produktnamen und Produkt-Identifizierer implementierter System-Komponenten
- eingesetzte Betriebssysteme, Middleware, Backend-Software, Software-Bibliotheken und interne Applikationen sowie deren Software-Versionen
- installierte Service-Packs, Patches, Hotfixes
- Seriennummern von Komponenten sowie hinterlegte Produkt-Lizenzen
- Datenbankstrukturen

Typische Beispiele für Ausgaben und Meldungen, in denen eine Preisgabe solcher System-Informationen potenziell auftreten kann:

- Anmelde-Fenster und -dialoge
- Fehlermeldungen
- Statusausgaben
- Banner von aktiven Netzwerk-Diensten
- System-Protokolle und Log Dateien
- Debug-Protokolle, Stack Traces

Soweit es technisch umsetzbar ist, ohne die Funktion und den Betrieb des Systems zu beeinträchtigen, muss die Ausgabe betroffener System-Informationen grundsätzlich deaktiviert sein.

Ein Zugriff auf betroffene System-Informationen darf nur dazu autorisierten Benutzern des Systems möglich sein. In der Regel ist der Kreis autorisierter Benutzer auf Administratoren und Betreiber des Systems zu beschränken. Ebenfalls zulässig ist ein Zugriff für autorisierte Monitoring- und Inventarisierungs-Systeme innerhalb der Einsatzumgebung.

Eine zulässige Ausnahme dieser Einschränkungen besteht für spezifische einzelne System-Informationen, deren Preisgabe für die vorgesehene Funktion des Systems in Verbindung mit dritten Systemen technisch zwingend erforderlich ist; beispielsweise die Präsentation unterstützter Protokolle und deren Versionen während der initialen Parameter-Aushandlung in Sitzungs-Aufbauten zwischen einem Client und einem Server.

Motivation: Informationen über die internen Strukturen eines Systems können von einem Angreifer genutzt werden, um Angriffe auf das System äußerst effektiv vorzubereiten. Beispielsweise kann ein Angreifer aus der Software-Version eventuell bekannte Schwachstellen eines Produktes ableiten, um diese gezielt während des Angriffs auf das System auszunutzen.

Umsetzungsbeispiel: [Beispiel 1]

Deaktivierung der Anzeige des Produktnamens und der installierten Version in den ausgelieferten Fehler-Webseiten eines eingesetzten Web-Servers.

[Beispiel 2]

Entfernung des Produktnamens und des zugehörigen Versions-Strings aus dem Login-Banner eines eingesetzten SSH Servers.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.01-9/9.0

Req 49	Alle Datenbankdienste müssen mit minimalen Rechten (least privilege principle) auf Betriebssystemebene aufgesetzt werden.
--------	---

Motivation: Durch die Verwendung von minimalen Rechten lässt sich die Gefahr der Systemkorrumpierung reduzieren.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-17/9.0

Req 50	Datenbankfunktionen, die den Zugriff auf Dateien des Betriebssystems erlauben, müssen gelöscht oder deaktiviert werden.
--------	---

Motivation: Einige Datenbanksysteme erlauben den Zugriff auf die Betriebssystemebene über spezielle stored procedures oder SQL-Funktionen. Es sind zahlreiche Schwachstellen dieser Prozeduren bekannt, die es einem Angreifer erlauben, unauthorisiert Zugriff zu einem System zu erlangen oder Schadcode auf dem Zielsystem auszuführen. Häufig sind diese Prozeduren mit unnötig hohen, zumeist privilegierten Rechten versehen oder erlauben jedem die Ausführung ohne Authentisierung. Malware nutzt zum Beispiel die Prozeduren MS-SQL: xp_cmdshell und Oracle: utl_tcp.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-20/9.0

Req 51	Datenbankfunktionen, die den Zugriff auf andere Netzdienste (z. B. SMTP, HTTP, SNMP, FTP etc.) erlauben, müssen gelöscht oder deaktiviert werden.
--------	---

Motivation: Datenbanksysteme stellen zum Teil Funktionen zur Verfügung, die üblicherweise ein Applikationsserver anbietet. So ist es zum Beispiel möglich, über spezielle Funktionen E-Mails zu versenden oder Webanfragen an externe Systeme zu starten.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-21/9.0

Req 52	Datenbankfunktionen, welche einen Zugriffe auf Betriebssystemebene und/oder Netzwerkdienste ermöglichen, muss das Zugriffsrecht der Rollen/Gruppen „Public“ und/oder „Everyone“ entzogen werden.
--------	--

Motivation: Durch den Entzug der Rechte lässt sich ein unautorisierter Zugriff auf Netzwerkdienste und die Betriebssystemebene durch die Nutzung von Datenbankfunktionen verhindern. Funktionen zur Ausführung von Betriebssystembefehlen und zur Nutzung von Netzwerkdiensten steht somit nicht allen, sondern nur autorisierten Benutzern zur Verfügung.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-22/9.0

Req 53	Die Betriebssystemrechte für die Dateien und Verzeichnisse der Datenbank (Programm-, Control-, Trace- und Logdateien) müssen exklusiv dem Betriebssystemkonto des Datenbanksystems zugeordnet werden.
--------	---

Teilweise speichern Datenbanksysteme sensible Account-Daten in Trace- oder Log-Dateien im Klartext ab. Daher müssen die Zugriffsrechte auf sensible Dateien und Verzeichnisse des Datenbanksystems so gesetzt sein, dass nicht-administrative Nutzer keine Lese-, Schreib- oder Ausführungsrechte besitzen. Der Zugriff zu wesentlichen Systemdateien und Verzeichnissen des Datenbanksystems muss dem Benutzer- konto des Datenbanksystems vorbehalten sein.

Motivation: Durch Umsetzung von restriktiven Zugriffsrechten lässt sich die Gefahr von Manipulationen deutlich reduzieren.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-23/9.0

Req 54	Datenbankzugriffe zwischen verschiedenen Datenbanksystemen müssen nach dem least privilege principle erfolgen.
--------	--

Die Zugriffsrechte müssen mit minimalen Privilegien ausgestattet sein. Der Benutzer darf nur die notwendigen Rechte zum Abfragen oder Extrahieren der erforderlichen Daten auf dem anderen Datenbanksystem haben. Nicht notwendige Rechte müssen dem Benutzer entzogen werden, damit er z. B. nicht unberechtigt andere Daten lesen oder verändern kann.

Motivation: Eine Beschränkung der Zugriffsrechte verkleinert die Angriffsfläche.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-24/9.0

Req 55	Zugriffe auf Datenbanksysteme sowie kritische Datenbank-Procedures und Datenbankinhalte müssen geloggt werden.
--------	--

Für einen sicheren und nachvollziehbaren Betrieb einer Datenbank ist das Loggen wichtiger Betriebsinformationen notwendig. Hierzu zählt zum Beispiel das Erfassen von erfolglosen Loginversuchen, um mögliche Einbruchversuche zu entdecken. Die Protokollierung sicherheitsrelevanter Benutzeraktionen muss in Übereinstimmung mit dem jeweils geltenden nationalen Recht erfolgen. Bei der Umsetzung von Maßnahmen, die sich aus dieser Anforderung ergeben, sind die jeweils geltenden Beteiligungsrechte der zuständigen Sozialpartner sowie die jeweils bestehenden betrieblichen oder tariflichen Regelungen zu beachten.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unbemerkt durchführbare Angriffe

ID: 3.16-26/9.0

Req 56	Datenbankzugriffe zwischen verschiedenen Datenbanksystemen müssen über einen individuellen Benutzeraccount erfolgen.
--------	--

Motivation: Die Zugriffe müssen über einen speziellen Benutzer erfolgen. Nur so lassen sich die Zugriffsrechte individuell auf das unbedingt notwendige Maß begrenzen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-25/9.0

Req 57	Wichtige Datenbank-Dienste und -Instanzen müssen kontinuierlich auf Missbrauchsszenarien hin überwacht werden.
--------	--

Die Überwachung von Benutzeraktionen auf Missbrauch muss in Übereinstimmung mit dem jeweils geltenden nationalen Recht erfolgen.

Es muss eine Fallstudie zur Art des Dienstes/Nutzerdatens, der Anzahl der Einträge und möglichen Missbrauchsszenarien geben, um das notwendige Maß an Überwachung der Fälle zu definieren.

- Insider-Bedrohungen
- Anomalieerkennung
- Accidental exposure
- Fehlkonfiguration
- Verwundbare Anwendung
- Machine Language Models
- Unsichere Daten im Transit
- ...

Motivation: Die missbräuchliche Nutzung von Datenbanksystemen ist vielfältig denkbar. Benutzer erzeugen eine ungewöhnlich hohe Datenlast beziehungsweise agieren zu ungewöhnlichen Uhrzeiten. Angreifer nutzen bei Datenbankabfragen unübliche und kritische Befehle sowie Tools und Schadcode, um ihre Rechte zu erweitern. Um eine missbräuchliche Nutzung zu erkennen, sollten Datenbanksysteme kontinuierlich auf Missbrauchsszenarien überwacht werden etwa durch Datenbanktrigger und Logüberwachung.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen

ID: 3.16-32/9.0

Req 58 Massendatenexfiltration muss erkannt werden, wenn mehr als 100.000 Kundendaten und DSK2 oder DSK3 vorhanden sind.

Die Massendatenexfiltrationserkennung ist eine kritische Kontrolle, die überwacht, ob ungewöhnlich große Datenmengen intern übertragen oder das Unternehmensnetzwerk oder die Datenbanksysteme verlassen. Wenn mehr als 100.000 Kundendaten – insbesondere als DSK2 oder DSK3 klassifiziert – abgerufen, kopiert oder übertragen werden, sollten automatisierte Alarmer ausgelöst werden. Diese schwellenwertbasierte Erkennung sollte mit SIEM- und User and Entity Behavior Analytics (UEBA)-Tools integriert werden, um Abweichungen von normalen Zugriffsmustern zu identifizieren. Das System sollte den Zugriffskontext (Benutzerrolle, Zeit, Datensensitivität und Übertragungskanal) korrelieren, um legitime Massenoperationen von potenziellen Exfiltrationsversuchen zu unterscheiden. Nach Entdeckung sollten vordefinierte Vorfalldiagnose-Workflows die Aktivität sofort isolieren, das Sicherheitsoperationszentrum informieren und forensische Beweise für die Untersuchung sichern.

Motivation: Kundendaten sind eines der wertvollsten Vermögenswerte des Unternehmens. Deshalb müssen wir alle Maßnahmen ergreifen, um massenhafte Datenlecks, böswillige Handlungen oder Missbrauch zu vermeiden. Dafür müssen wir wissen, wie wir unsere Datensätze handhaben und nutzen.

Umsetzungsbeispiel: Technische Leitlinien und Implementierungsdetails werden ebenfalls auf dieser Seite bereitgestellt und kontinuierlich gepflegt:

<https://yam-united.telekom.com/pages/psa/apps/wiki/wiki-sicherheitsanforderungen/list>

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Modifikation von Daten
- Angriffe motiviert und erleichtert durch Informationspreisgabe oder erkennbare Sicherheitsschwachstellen

ID: 3.16-33/9.0

2.7. HDFS Configuration

Req 59 For HDFS, a umask of 077 must be set.

While any access to data in the HDFS should be controlled by a central security solution anyway, it is good practice to implement the security principle of “defense in depth” by setting according file permissions. In case Apache Ambari is used, such setting can be performed by setting “Service” -> “HDFS” -> “Advanced hdfs-site” and setting “fs.permissions.umask-mode = 077”.

ID: 3.98-59/2.0

Req 60 The system user used to run the HDFS daemon must not be used for any other service.

The system user used to run the HDFS daemon must not be used for any other service.

ID: 3.98-60/2.0

Req 61 Fallback for authentication to local authentication mechanisms must be disabled.

Fallback for authentication to local authentication mechanisms must be disabled. In case Apache Ambari is used, such setting can be performed by setting “Service” -> “HDFS” -> “Advanced ranger-hdfs-security” and setting

“xasecure.add-hadoop-authorization=false”.

ID: 3.98-61/2.0

2.8. Kerberos Configuration

Req 62	Kerberos must be used as a basis for identification, authentication, and authorization; i.e., Hadoop “kerberos” authentication mechanisms must be used.
--------	---

Identification, authentication, and authorization are essential parts of granting access to data. As Kerberos is the supported service to do so by Hadoop, it must be used.

In Hadoop available is also the “simple” mechanism, which is insufficient in regard to security for production environments. It might be an option on small test or development systems if only synthetic data is used; in such case, even pseudo- or anonymized data must not be used.

ID: 3.98-62/2.0

Req 63	The KDC must be installed on two or more servers, i.e. in a master/slave configuration.
--------	---

As Kerberos is essential for security functions within a Hadoop installation, it must be configured on two or more servers to allow for redundancy.

ID: 3.98-63/2.0

Req 64	/etc/krb5.conf and /var/kerberos/krb5kdc/kdc.conf must contain installation-specific values allowing for secure operation.
--------	--

Those files can be updated on the KDC host, krb5.conf can then be copied to all other servers. The Deutsche Telekom Security Requirement Cryptographic Algorithms and IP Security (IPSec), must be honored. Care must be taken to remove potentially insecure cryptographic algorithms, cp. “supported encyptes” below. For example, one might need to delete “RC4” from this list.

Umsetzungsbeispiel: Example of a krb5.conf:

```
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log[libdefaults]
dns_lookup_realm = false
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true
rdns = false
default_realm = DATA.LAKE
default_ccache_name = KEYRING:persistent:%{uid}[realms]
DATA.LAKE = {
  kdc = kdc.data.lake
  admin_server = kdc.data.lake
}[domain_realm]
.data.lake = DATA.LAKE
data.lake = DATA.LAKE
```

Example of a kdc.conf:

```
[kdcdefaults]
kdc_ports = 88
kdc_tcp_ports = 88[realms]
DATA.LAKE = {
  master_key_type = aes256-cts
  acl_file = /var/kerberos/krb5kdc/kadm5.acl
  dict_file = /usr/share/dict/words
```

```
admin_keytab = /var/kerberos/krb5kdc/kadm5.keytab
supported_encetypes = aes256-cts:normal aes128-cts:normal des3-hmac-sha1:normal arcfour-hmac:normal camellia256-cts:normal camellia128-cts:normal
default_principal_flags = +preauth
max_renewable_life = 7d 0h 0m 0s
}
```

ID: 3.98-64/2.0

Req 65	Every user of a Hadoop-service and every service on each host used by a Hadoop user must have a unique Kerberos user principal name (UPN) and service principal name (SPN), resp.
--------	---

The same services running on different hosts must use different SPNs, because if credentials on one node are compromised, other nodes are not affected, and because Kerberos falsely detect reply attacks if by change different services with the same SPN request and use tickets at the same time.

ID: 3.98-65/2.0

Req 66	The FQDN must be appended to a service principal's name.
--------	--

Adding the FQDN to a service principal's name ensures uniqueness.

ID: 3.98-66/2.0

Req 67	For each Hadoop installation, a different Kerberos realm must be used.
--------	--

While it is technically possible to use the same Kerberos realm for production, test, and development installations, administrators and user will sooner or later assume they are working in e.g. a test installation, while actually they are working in the production setup. Such mistakes can mitigate by using different realm names for different installations.

ID: 3.98-67/2.0

Req 68	All UPNs and SPNs must be registered in a network-based directory service such as Active Directory or OpenLDAP.
--------	---

All users of a cluster must be provisioned on servers of a cluster. Two options are available to do so: Provisioning all users locally, e.g. in the `/etc/passwd`, or using a network-based directory service. In practice, consistency of accounts can only be ensured if using network-based directory services.

ID: 3.98-68/2.0

Req 69	Non-administrative users must not be enabled for shell access.
--------	--

Regular users must not utilize a system's shell access. Best practice on Unix-like systems is to assign each such the default shell `/sbin/nologin` and to use `AllowUsers`, `DenyUsers`, `AllowGroups`, and `DenyGroups` settings in `/etc/ssh/sshd_config`.

ID: 3.98-69/2.0

Req 70	Keytab files must be accessible only by the system user running the respective service. Such access must only be read access.
--------	---

File permissions of keytab files must be set to 400, the owner of the keytab file must be the system owner running the respective service.

As the keytab files contain credentials unencrypted, those files must be secured.

ID: 3.98-70/2.0

Req 71	„Java Cryptographic Extension“ must be installed on all nodes.
--------	--

This requirement is a hint for administrators setting up a Hadoop installation. This package is required for cluster operation.

ID: 3.98-71/2.0

2.9. Knox Configuration (Hortonworks)

Req 72	For all but administrative access, Knox must be used for edge nodes allowing access to data in the Data Lake.
--------	---

The Apache Knox Gateway („Knox“) is a system to extend Hadoop services to users outside of a Hadoop installation while ensuring enforcement of security policies. Further, Knox simplifies security for users who access data from the Data Lake and who execute jobs.

Knox simplifies access by extending Hadoop's REST/HTTP services by encapsulating Kerberos.

Knox enhances security by exposing Hadoop's REST/HTTP services without revealing Hadoop installation network details and by providing TLS out of the box.

Knox centralizes controls by central enforcement of REST API security.

Knox integrates with Hadoop's Kerberos installation, which in turn integrates with a company's identity management solutions like Active Directory.

As of the time of writing, Knox supports the following Hadoop services, subject to specific versions:

- YARN
- WebHDFS
- WebHCat/Templeton
- Oozie
- HBase/Stargate
- Hive (via WebHCat)
- Hive (via JDBC)

ID: 3.98-72/2.0

Req 73	The security directory, \$gateway/data/security, and its contents must be readable and writeable only by the operating system user running the Knox component.
--------	--

In this directory the master secret and other confidential information is stored. Therefore, care must be taken to ensure security of this data.

ID: 3.98-73/2.0

Req 74	Web application security filters available for Knox must be installed and used.
--------	---

As of the time of writing, one filter addressing cross site request forgery (CSRF) exists.

ID: 3.98-74/2.0

Req 75	All requests to Knox must be documented to allow for both developers knowing how to access Knox and administrators of possibly used web application firewalls to configure their systems.
--------	---

As of the time of writing, one filter addressing cross site request forgery (CSRF) exists.

ID: 3.98-75/2.0

Req 76	Auf einer Betriebssysteminstanz (mit physikalischer oder virtualisierter Hardware) dürfen nur dann mehrere Datenbankinstanzen zusammen betrieben werden, wenn diese • den gleichen, niedrigen Schutzbedarf haben, • unter einer einzigen Kundenhoheit stehen und • administrativ vom gleichen Personenkreis betrieben werden.
--------	---

Motivation: Durch den gemeinsamen Betrieb mehrerer Datenbankinstanzen steigt die Gefahr, dass Benutzer unberechtigterweise Zugriff auf Instanzen erhalten, für die sie weder fachlich noch administrativ zuständig sind.

Umsetzungsbeispiel: Zur Trennung der einzelnen Datenbankinstanzen können Administratoren physikalische Serverhardware, wie auch Virtualisierungslösungen einsetzen.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System
- Unberechtigte Einsichtnahme oder Abgriff von Daten
- Unberechtigte Nutzung von Diensten oder Ressourcen
- Unbemerkt durchführbare Angriffe

ID: 3.16-9/9.0

Req 77	(Default-)Passwörter auf Datenbanksystemen müssen geändert werden.
--------	--

Motivation: Viele Datenbankhersteller versehen bei der Installation verschiedene Benutzer mit (Default-)Passwörtern (sa/blank; DBSNMP/DBSNMP, etc.). Ein Angreifer kann das Vorhandensein dieser (Default-)Passwörter ausnutzen, um zumeist auch privilegierten Zugriff auf das Datenbanksystem zu erhalten.

Für diese Anforderung sind folgende Bedrohungen relevant:

- Unberechtigter Zugang zum System

ID: 3.16-16/9.0