Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Trusted Computing

Prof. Dr. (TU NN)
Norbert Pohlmann

Institut für Internet-Sicherheit – if(is)
Westfälische Hochschule, Gelsenkirchen
http://www.internet-sicherheit.de

if(is)
internet-sicherheit.

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

- **Ziele und Ergebnisse der Vorlesung**
- Einleitung
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Ziele und Ergebnisse der Vorlesung
→ Trusted Computing

- Gutes Verständnis über die **IT-Sicherheitsarchitektur** und **IT-Sicherheitsprinzipien** von Trusted Computing erlangen.
- Erlangen der Kenntnisse über die *TPM Schlüsselhierarchie*, **Authenticated Boot**, **Attestation**, **Binding**, **Sealing** und *Trusted Network Connect (TNC)*.
- Gutes Verständnis zu verschiedenen **Kernelarchitekturen** erlangen.
- Gutes Verständnis über praktische Anwendungen mit Hilfe der IT-Sicherheitsplattform **Turaya**.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- **Einleitung**
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Definition von Vertrauenswürdigkeit

- **Ein IT-System ist vertrauenswürdig, wenn es sich immer in der erwarteten Weise für den beabsichtigten Zweck verhält.**

- Wie kann das überprüft werden?
  - Der Nutzer macht seine **eigenen Erfahrungen** mit der erwarteten Weise für den beabsichtigten Zweck des IT-Systems

    oder

  - Der Nutzer **vertraut jemandem**, der Referenzen für die erwartete Weise des beabsichtigten Zweck des IT-Systems bereitstellt *(Vertrauenspersonen, Reputationssysteme, Zertifizierungen, …)*

    oder

  - **Mit Trusted Computing Funktionen**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Idee

- **Trusted Computing Group (TCG):**
  Industriekonsortium bestehend aus den führenden IT-Firmen
  (Hewlett-Packard, IBM, Intel, AMD, Microsoft, Sony, Sun, Infineon, ...)

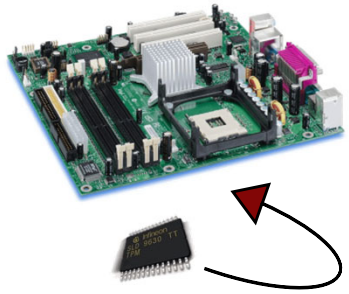
- **Grundmotivation**
  - Entwicklung **offener Spezifikationen** für **vertrauenswürdige IT-Systeme** (Server, PC, eingebettet, usw.)
  - Sicherheit verteilter Anwendungen mit wirtschaftlich vertretbarem Aufwand verbessern
  - Keine massive Veränderung existierender Hard- bzw. Software

- **Hauptidee**
  - Manipulationssichere Komponente in Hardware (sicherer als Software)
    → Stärkung gegen Software-basierte Angriffe
  - Sicherheit des IT-Systems reduziert auf die Sicherheit eines Moduls
  - Integrität und Authentizität eines IT-Systems zuverlässig überprüfbar, auch aus der Distanz

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Funktionen (1/2)

- **Trusted Platform Modules (TPM)**
  - Sicherer Zufallsgenerator (sichere kryptographische Schlüssel)
  - Kryptographische Funktionen: Signatur (RSA), Hash (SHA-X)
  - Erzeugung verschiedener kryptographischer Schlüssel
  - **Platform Configuration Register (PCR)**
    → Zur Speicherung der Systemkonfiguration



- **Sicherer Speicher**
  - **Erzeugung sicherer kryptographischer Schlüssel** und
  - **sichere Speicherung von Schlüssel** im Hardware-Sicherheitsmodul (TPM)

- **Sealing (versiegeln)**
  - **Kryptographische Schlüssel** können an das **IT-System** und/oder eine bestimmte **Softwarekonfiguration** gebunden werden
    → Schutz vor Manipulationen des Betriebssystems

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Funktionen (2/2)

- **(Remote) Attestation**
  - Aktuelle Systemkonfiguration des IT-Systems wird dargestellt
  - Erkennung manipulierter IT-Systeme (Verteilte Systeme, Web-S.)
  - Kommunikation nur mit vertrauenswürdigen IT-Systemen

- **Access Control**
  - Durchsetzung von Zugriffsregeln in einem Netzwerk mit unbekannten IT-Systemen (TNC)

- **Überprüfbares Booten**
  - Systemkonfiguration kann überprüft werden, z.B. mittels eines persönlichen Gerätes (Smarcard, USB-Stick, Handy, …)

- **Verbreitung von TPMs**
  - Da IT-Systeme wie Notebooks fast ausschließlich **„Microsoft Ready“** sind, ist ein **TPM auf den meisten der IT-Systeme verfügbar**.

TPM

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Digitalisierung
→ Herausforderungen

- **Das Software-Problem**

- **Malware auf zu vielen IT-Systemen**

Durch eine starke Isolierung den Schaden begrenzen

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Cyber-Sicherheitssysteme
→ Reaktive Cyber-Sicherheitssysteme

- Bei den heutigen **reaktiven Cyber-Sicherheitssystemen**, wie Anti-Spam-, Anti-Malware-, Intrusion-Detection-Systemen, ist die **grundsätzliche Idee**, so **gut** und **schnell** wie möglich **Cyber-Angriffe zu erkennen**.

- Das bedeutet, wenn die Cyber-Sicherheitslösungen einen Angriff durch eine entsprechende **Angriffssignatur** oder eine **Anomalie** erkennen, dann wird versucht, das IT-System so schnell wie möglich zu schützen, um den Schaden zu reduzieren.

„Airbag-Methode“
Wenn’s passiert, soll es weniger wehtun!

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Cyber-Sicherheitssysteme
→ Proaktive Cyber-Sicherheitssysteme

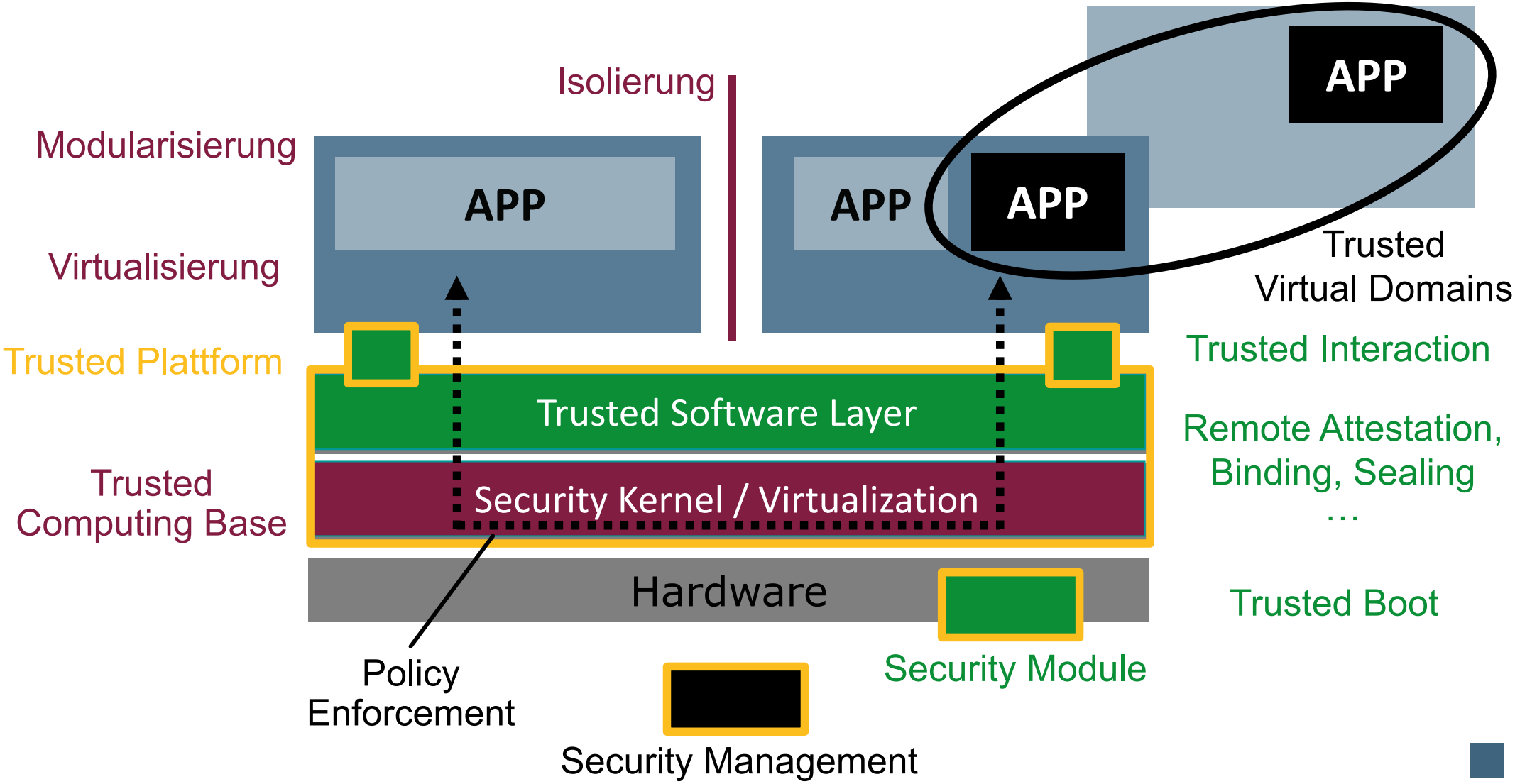
- Für die zunehmende **Vielfalt und Komplexität** der IT-Endgeräte, IoT-Geräte, Netzkomponenten und IT-Infrastrukturen werden deutlich **verlässlichere**, **robustere** und **wirkungsvollere Cyber-Sicherheitskonzepte** benötigt.
- Daher ist es sinnvoll, weniger reaktive und mehr **moderne proaktive Cyber-Sicherheitssysteme** zu verwenden.
- Proaktiven Cyber-Sicherheitssysteme arbeiten mit einem kleinen **Sicherheitskern** (sichere Betriebssysteme) und **Virtualisierung**, können Software messbar machen und mit einer **starken Isolation** Anwendungen mit ihren Daten separieren und so nachhaltige und angemessene Cyber-Sicherheit bieten.
- Für proaktive Cyber-Sicherheitssysteme muss die Softwarearchitektur der IT-Endgeräte allerdings grundlegend anders aufgebaut sein als bisher.

„ESP-Strategie“
Verhindern, dass ein „Auto“ überhaupt ins Schleudern kommt

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ IT-Sicherheitsprinzipien

Robustness/Modularity **Trusted Process** **Integritätsprüfung**



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

TC-Sicherheitsaspekte
→ Virtualisierung

- Ein wichtiger IT-Sicherheitsaspekt ist die Virtualisierung auf dem Endgerät.
- Der Vorteil von Virtualisierung besteht darin, dass **auftretende Fehler** (Schwachstelle, Malware, …) im Prinzip in einer virtuellen Maschine in einem **abgeschlossenen Bereich begrenzt bleibt** und nicht eine andere virtuelle Maschine infizieren kann.
- Es ist auch sehr einfach möglich, die verschiedenen virtuellen Maschinen wieder in einen **stabilen Urzustand** zu versetzen und von da aus neu zu starten.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

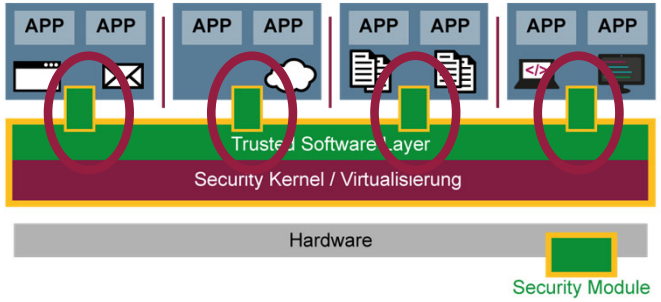
TC-Sicherheitsaspekte
→ Isolierung

- Der IT-Sicherheitsaspekt **Isolierung** sorgt dafür, dass die **virtuellen Maschinen** zusätzlich weiter stark isoliert und sicher getrennt voneinander laufen und **sich nicht gegenseitig beeinflussen** können.

- Aus diesem Grund haben z.B. Schwachstellen und Malware in einer isolierten virtuellen Maschine **keinen Einfluss auf die anderen virtuellen Maschinen** hat.

- Eine solche stark isolierte virtuelle Maschine wird im Bereich von TC auch Compartment genannt.

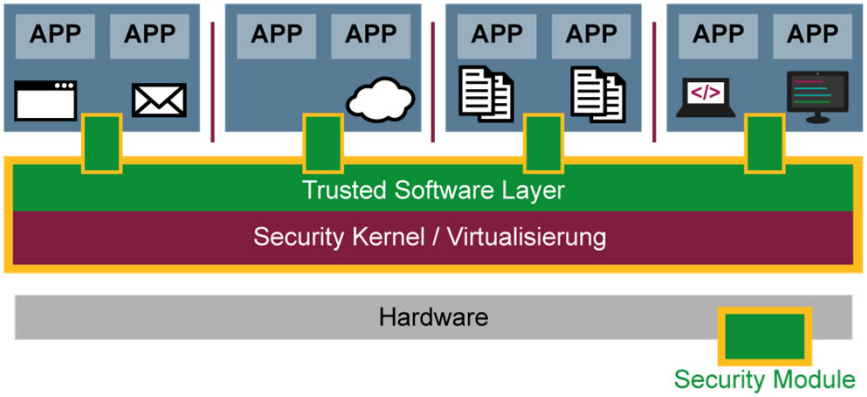
- Methoden der Isolierung:
  - **Enforcement der Kommunikation**
  - individuelle **Verschlüsselung der Daten** in den virtuellen Maschinen



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

TC-Sicherheitsaspekte
→ Modularisierung

- Der IT-Sicherheitsaspekt der Modularisierung ist eine **Möglichkeit**, **Anwendungen**, die **zusammen** gehören, in einer virtuellen Maschine laufen zu lassen und Anwendungen, die **getrennt** sein sollten, **in verschiedenen virtuellen Maschinen** zu positionieren.
- Dieser IT-Sicherheitsaspekt offeriert einen interessanten **Gestaltungsspielraum**, mit dem eine sehr hohe Cyber-Sicherheit erzielt werden kann, weil für **verschiedene IT-Sicherheitslevel** von Anwendungen unterschiedliche virtuelle Maschinen genutzt werden können.



- **Eine Beispiel ist:**
  Das **Office-Paket** läuft in einer virtuellen Maschine, das **Design-Paket** für die **Business-Anwendung** in einer anderen und der **Browser** hat auch eine separate virtuelle Maschine.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

TC-Sicherheitsaspekte
→ Trusted Boot/Authenticated Boot

- Mithilfe von **Trusted Boot** oder **Authenticated Boot** kann dafür gesorgt werden, dass ein IT-System nur in einem **definierten vertrauenswürdigen Zustand** aktiv wird.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

TC-Sicherheitsaspekte
→ Remote Attestation

- **Remote Attestation** gibt die Möglichkeit, die **Vertrauenswürdigkeit** von **anderen, auch fremden IT-Systemen zu messen**, bevor eine Interaktion mit diesem IT-System begonnen wird.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

TC-Sicherheitsaspekte
→ Binding/Sealing

- Binding und Sealing sind weitere Trusted Computing-Funktionen, mit denen moderne IT-Sicherheitssystem intelligent und vertrauenswürdig umgesetzt werden können.

- Bei **Binding** werden **verschlüsselte Daten an ein TPM gebunden**.

- Bei **Sealing** werden **verschlüsselte Daten an die Software- und Hardware-Konfiguration eines IT-Systems** sowie an ein TPM **gebunden**.

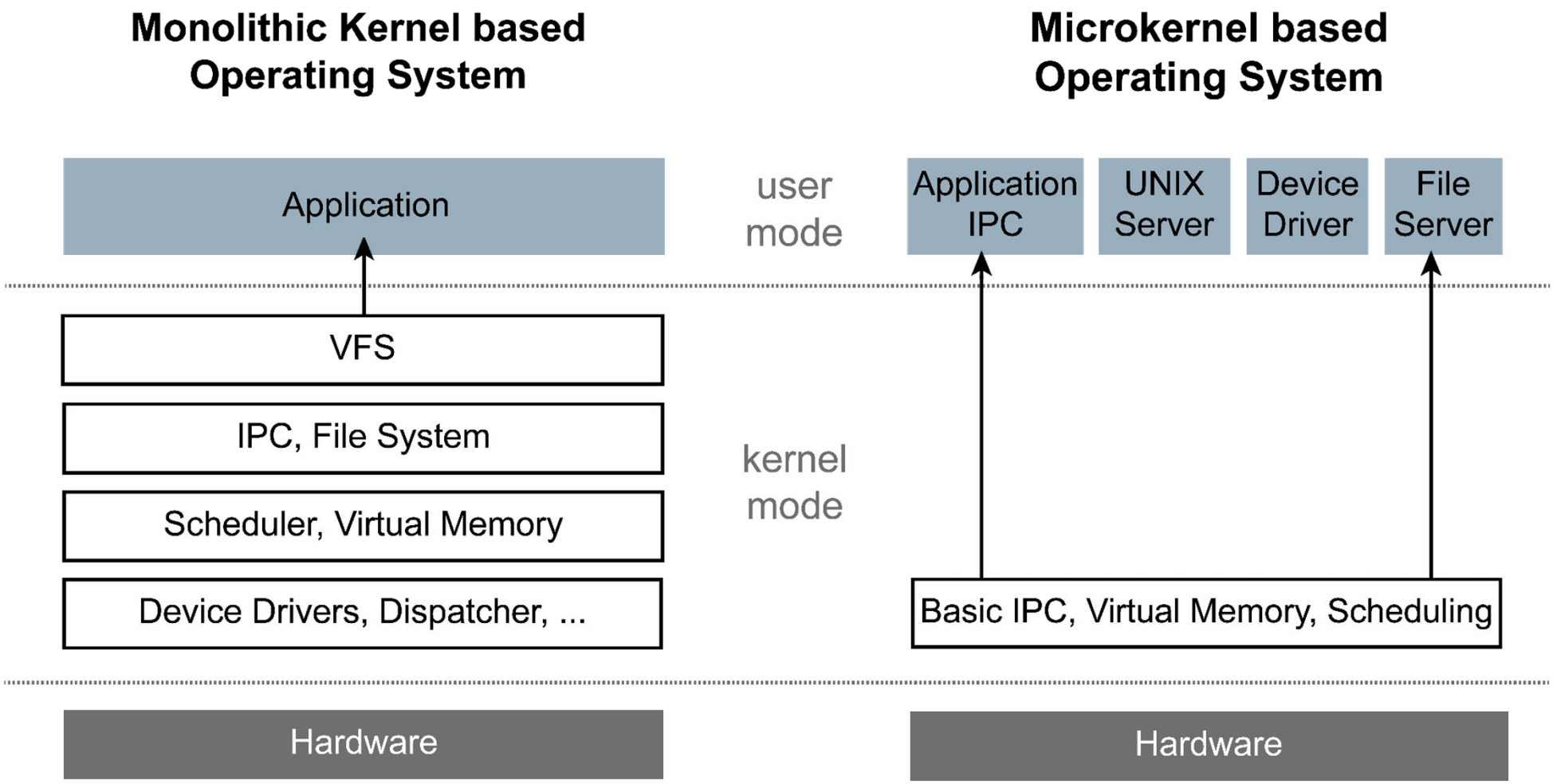
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- **IT-Sicherheitsarchitektur**
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Kernelarchitekturen (1/3)



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Kernelarchitekturen (2/3)

- **Vorteile** eines monolithischen Kernels:
  - Lange etabliert
  - Gute Performance

- **Nachteile** eines monolithischen Kernels:
  - Alle Treiber vereint im Kernel-Space
  - Geringere Flexibilität
  - Höhere Komplexität
  - Wenig robust
  - Schlechte Sicherheitsmechanismen

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Kernelarchitekturen (3/3)

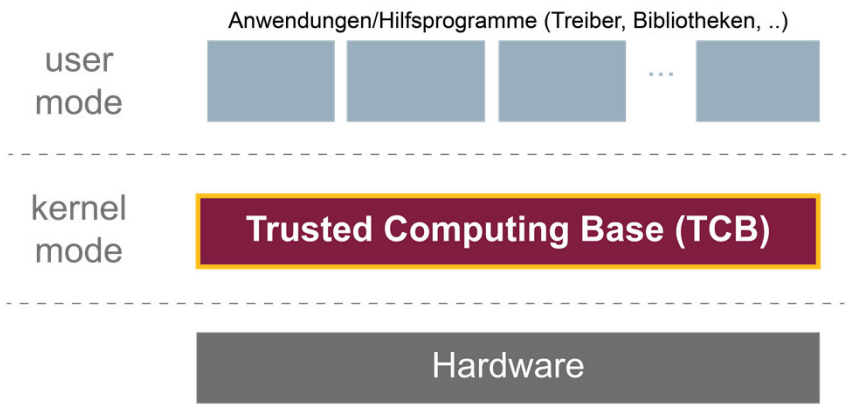
- **Vorteile** eines Mikrokernels:
  - Höhere Robustheit
  - Höhere Modularität
  - Höhere Flexibilität
  - Höhere IT-Sicherheit (kontrollierbare Interprozesskommunikation)
  - Weniger benötigter Speicherplatz

- **Nachteile** eines Mikrokernels:
  - Weniger Leistung durch mehr Kommunikation zwischen den Prozessen

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Trusted Computing Base (TCB) – 1/2

- Eine *Trusted Computing Base* dient als verlässliches **IT-Sicherheitsfundament**, um darauf weitere IT- und IT-Sicherheitskomponenten aufzubauen.



- Per Definition ist daher die **„Trusted Computing Base“ der kritische Teil eines IT-Systems**.

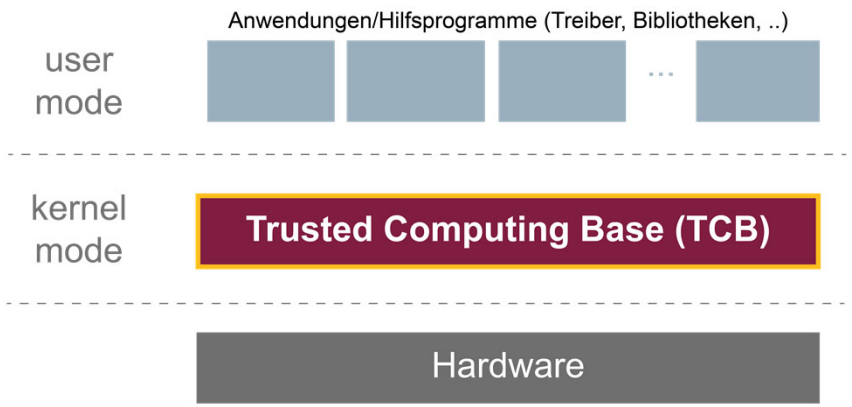
- Wenn im TCB eine **Schwachstelle** vorhanden ist, dann ist **das ganze IT-System kompromittierbar.**

- Wenn **außerhalb** der TCB eine Schwachstelle vorhanden ist, dann kann anhand einer **IT-Sicherheitspolicy** der **potenzielle Schaden** sehr eingeschränkt und **klar beschrieben** werden.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Trusted Computing Base (TCB) – 2/2

- Aus diesem Grund ist eine TCB **sehr sorgfältig designt und implementiert.**



- Eine auf einem Mikrokernel (Security Kernel) basierende TCB hat ca. 20.000 Lines of Code und ist von daher eine sehr **vertrauenswürdige Basis**, die in der Regel auch schon semiformal oder **formal verifiziert werden kann**.

- Mithilfe der formalen Beweisbarkeit wird eine **Sicherheitsevaluation auf hohem Niveau** möglich.

- Es gibt aber auch TCBs, die zum Beispiel aus einem sehr abgespeckten und speziell gehärteten Linux bestehen, das auch schon sehr viel vertrauenswürdiger sind als übliche Betriebssysteme.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Security Module
→ Trusted Platform Module (TPM)

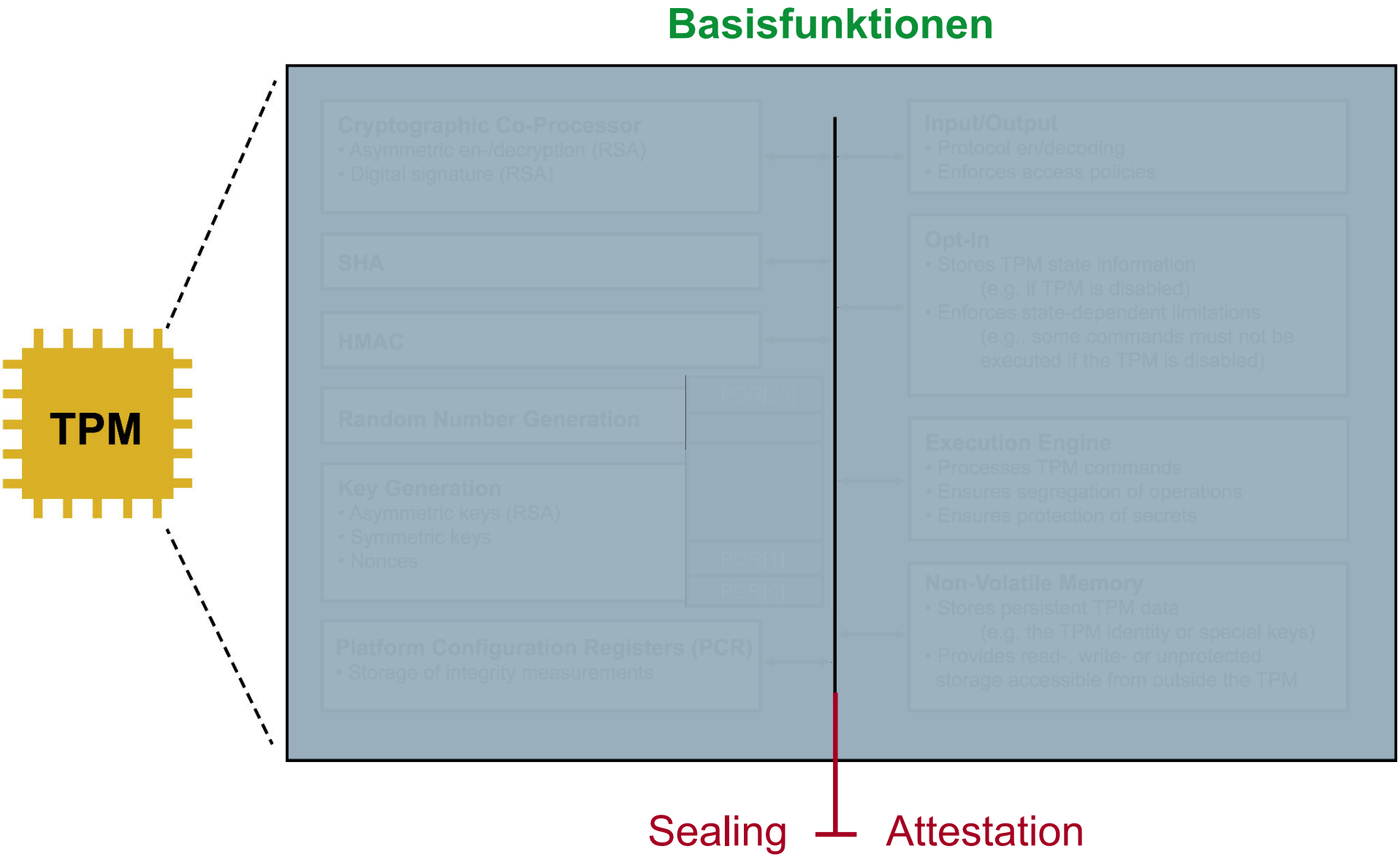
- Das Security Module ist z.B. ein TPM mit intelligenten kryptografischen Verfahren auf dem Level von Smartcard-Sicherheit, aber auch weiteren IT-Sicherheitsdiensten, wie die **Platform Configuration Register (PCR)**, die die sichere Speicherung und Überprüfung von Messdaten sicherstellt.
- Das TPM ist ein kleiner passiver Hardware-Sicherheitschip, der fest mit dem Mainboard verbunden ist.

Vorteile eines TPMs

- Die Hardware-Sicherheitsmodule bieten eine **sehr hohe IT-Sicherheit** bei **geringer Investitionssumme**, da ein TPM nicht mehr als ein Euro kostet.
- Die Hardware-Sicherheitsmodule sind schon auf den meisten IT-Systeme verfügbar, das heißt, die flächendeckende Einführung einer Sicherheitsplattform ist einfach! *(Alle IT-System, die „Microsoft Ready“ sind, müssen ein TPM verbaut haben.)*
- Das **TPM** ist in eine **Sicherheitsinfrastruktur (PKI, …) eingebunden** und daher einfach im Sicherheitsmanagement zu behandeln.

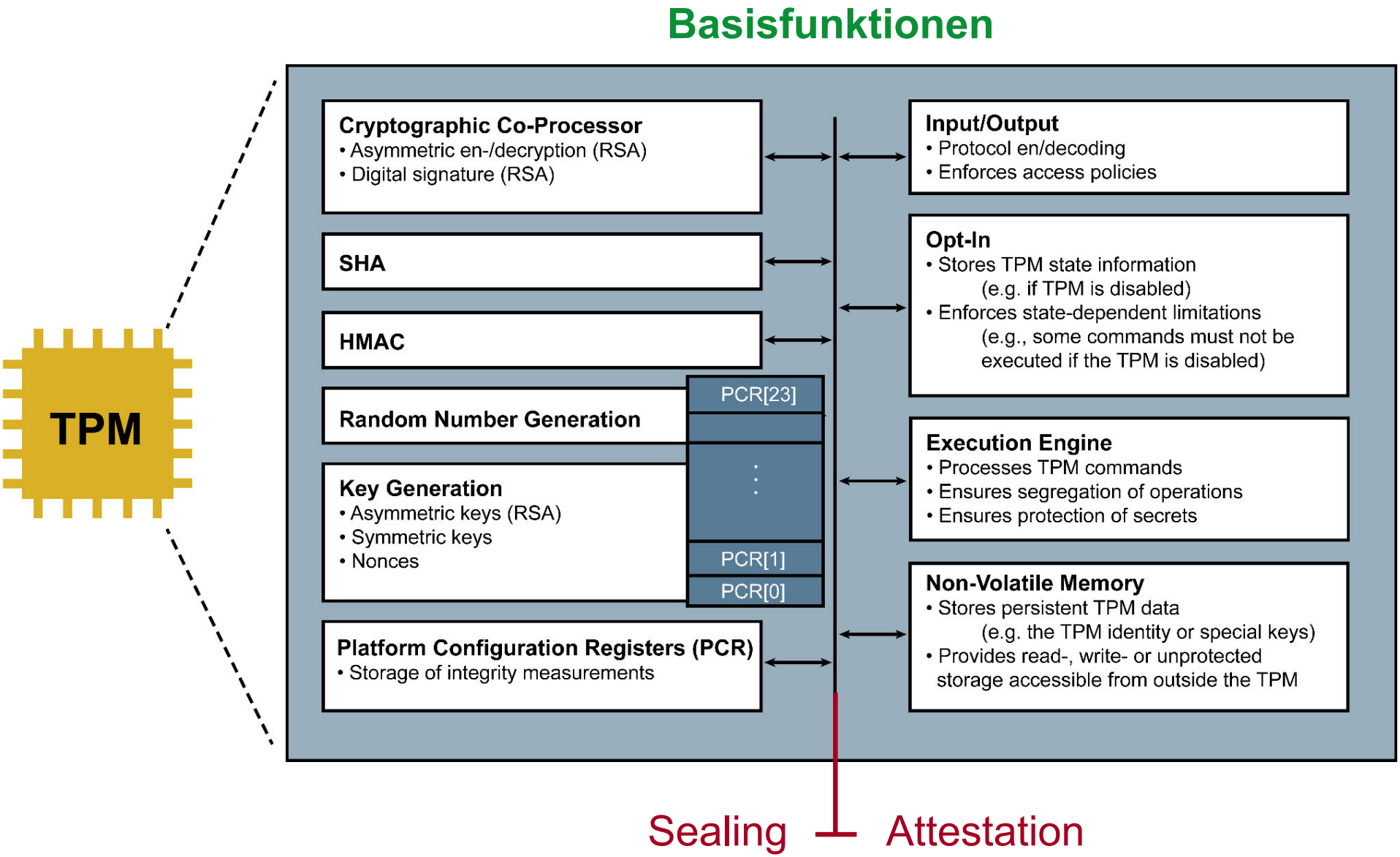
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

HSM: Trusted Platform Module
→ Hardware-Sicherheitsanker



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

HSM: Trusted Platform Module
→ Hardware-Sicherheitsanker



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur

→ Core Root of Trust for Measurement (CRTM)

- Eine **vertrauenswürdige Basis**, eine Trusted Platform, braucht eine Wurzel oder **Basis des Vertrauens**.
- Die Lösung bei der „Trusted Computing Group“ heißt **„Basis Root of Trust“**, die eine **Kette des Vertrauens** (Chain of Trust) bildet.
- Für normale IT-Systeme wurde von der Trusted Computing Group die Komponente **„Core Root of Trust for Measurement (CRTM)“** spezifiziert.
- Der CRTM ist dabei eine Software, die einen Messvorgang über einzelne Systemzustände (Hard- und Software) außerhalb der Trusted Platform vertrauenswürdig durchführt und dann die Ergebnisse innerhalb der Trusted Platform in die Platform Configuration Register (PCR) des TPMs hinterlegt.
- Die Ausführung der CRTM Software beginnt mit dem Bootvorgang.
- Technisch wird die **CRTM Software** in der Regel in das **BIOS** des IT-Systems integriert.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Idee von transitiven Vertrauen

Entity E_0

CRTM

Core Root of
Trust for Measurement

"Transitives Vertrauen„

- **Vertrauen ist transitiv** von E_0 nach E_1 nach E_2 bis E_n.
- Das Vertrauen von E_2 erfordert, dass E_0 und E_1 vertraut werden kann.
- Bei dem Konzept „Chain of Trust for Measurement“ ist das Ziel, das Vertrauen in Entität E_n zu gewinnen.
- Der Ablauf ist so: E_0 startet E_1, E_1 startet E_2 usw.
- Um E_n zu vertrauen, muss E_{n-1} vertraut werden. Um E_{n-1} zu vertrauen, muss E_{n-2} vertraut werden usw.
- E_0, E_1 bis E_n schaffen eine **„Vertrauenskette“**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Authenticated Boot / Secure Boot

- **Authenticated Boot:**
  - Systemzustände messen.
  - Speicherung in den PCRs.
  - Überprüfung der Integrität.
- **Secure Boot:**
  - Systemzustände messen.
  - Überprüfung der Integrität.
  - **Ggf. Bootvorgang stoppen.**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Identitäten (1/2)

- **Endorsement Key (EK):**
  - Eindeutige TPM-Identität (nicht migrierbar).
  - RSA-Schlüsselpaar (im Herstellungsprozess erzeugt).
  - Geheimer Schlüssel im TPM gespeichert.
  - Öffentlicher Schlüssel ist datenschutzsensitiv.
  - TPM-Hersteller verwaltet PKI.

- **Endorsement Credential (EC):**
  - Elektronisches Zertifikat vom TPM-Hersteller.
  - Bestätigt ordnungsgemäße Erstellung und Einbettung des EK.
  - Bestandteile: TPM-Herstellername, TPM-Modellnummer, TPM-Version, Öffentlicher Schlüssel des EK.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Identitäten (2/2)

- **Platform Identität (PI):**
  - Entspricht der TPM-Identität (EK).
  - Physikalische oder logische Bindung des TPMs an die Plattform (z.B. mittels anlöten an das Motherboard oder Kryptographie).
  - Plattform ≙ Motherboard/IT-System.
  - Plattform muss konform zur Evaluierungsrichtlinien der TCG sein → Conformance Credential (CC)
- **Platform Credential:**
  - Elektronisches Zertifikat vom Plattform-Hersteller.
  - Bestätigt gültige Verbindung zwischen TPM und Plattform → Trusted Plattform.
  - Bestandteile: Name des Plattformherstellers, Plattformmodell und Versionsnummer, Verweise auf die EC und CC.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (1/7)

- **Migratable Keys** → Auf andere Plattformen übertragbar (migrierbar).
- **Non-Migratable Keys** → An die Plattform gebunden (nicht migrierbar).

- **Storage Root Key (SRK):**
  - Wurzel der Schlüsselhierarchie.
  - Während der Installation des TPM-Eigentümers generiert.
  - Löschung des TPM-Eigentümers → Löschung des SRK
    → Kein Zugriff auf die Schlüsselhierarchie mehr.
  - **Eigenschaften**:
    - Steht im nicht flüchtigen Speicher des TPMs.
    - Ist nicht migrierbar.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (2/7)

- **Attestation Identity Keys (AIK):**
  - Verwendet für die Trusted Computing Funktion „**Attestation**“:
    - Authentische Bestätigung der Integrität einer Plattformkonfiguration (z.B. aktuelle Hard- und Softwareumgebung).
  - Nötig, da EK datenschutzsensibel ist.
  - AIKs werden vom TPM-Besitzer generiert.
  - TPM/Plattform kann mehrere AIKs besitzen (z.B. für Online-Banking, E-Mail, …)
  - **Eigenschaften:**
    - Stehen im nicht flüchtigen Speicher des TPMs.
    - Nicht migrierbar.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (3/7)

- **Storage Keys (StorK):**
  - Verschlüsselung von weiteren Schlüsseln und Daten außerhalb des TPMs.
  - Verwendet für die Trusted Computing Funktion „**Sealing**“:
    - Zustand der Plattform wird Teil der Verschlüsselung.
    - Entschlüsselung nur im vorher **definierten Zustand** (SW/HW) möglich.
  - **Eigenschaften:**
    - RSA-Schlüsselpaar.
    - Im Allgemeinen darf die Migration zu anderen TPMs erfolgen.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (4/7)

- **Binding Keys (BindK):**
  - Verschlüsselung von beliebigen Daten außerhalb des TPMs.
  - Entspricht der asymmetrischen Verschlüsselung.
  - **Eigenschaften:**
    - RSA-Schlüsselpaar (es können auch andere Algorithmen vom TPM unterstützt werden).
    - Im Allgemeinen darf die Migration zu anderen TPMs erfolgen.
    - Kann nur mit Binding-Befehlen verwendet werden.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (5/7)

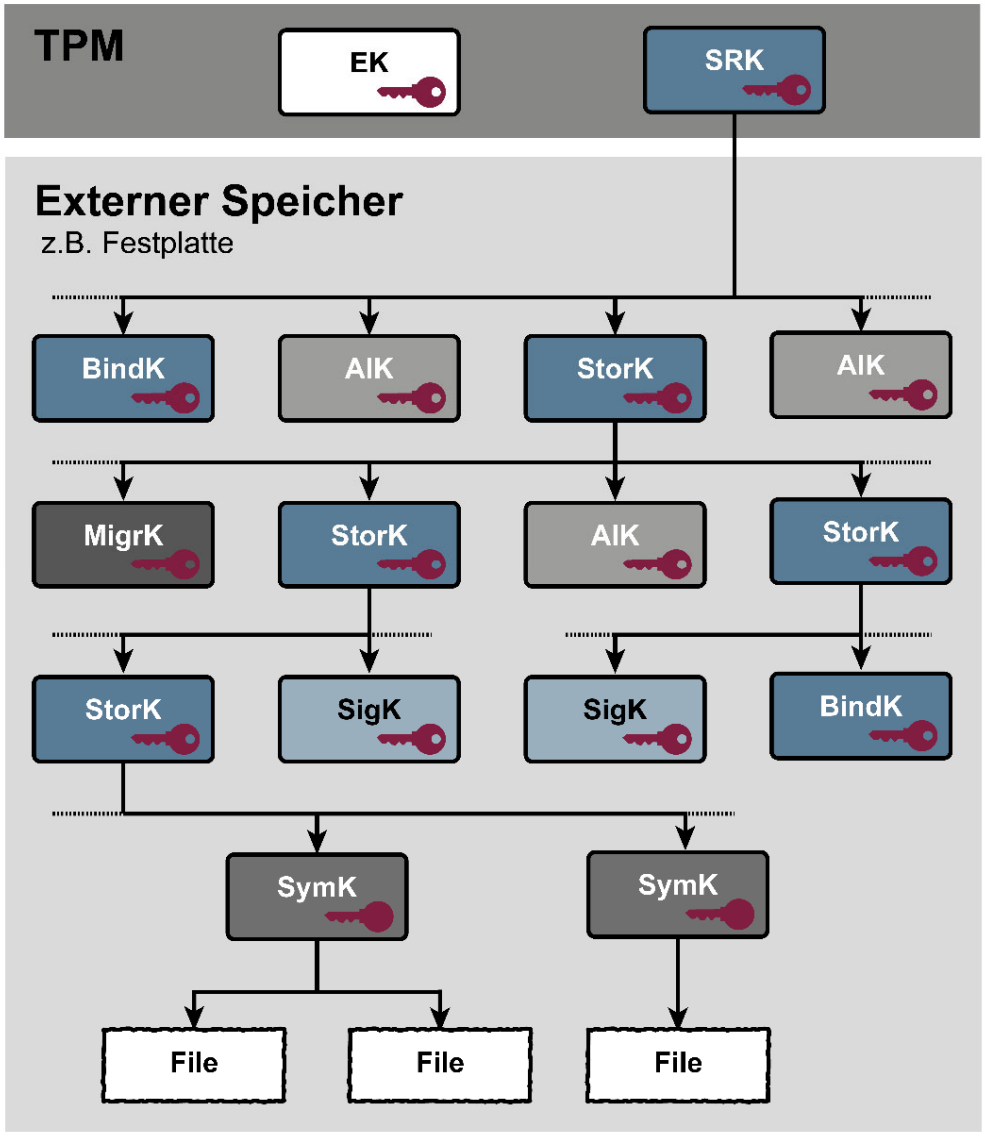
- **Signing Keys (SigK):**
  - Nachweis der Authentizität und Integrität von beliebigen Daten /Protokollnachrichten innerhalb und außerhalb des TPMs.
  - **Eigenschaften:**
    - RSA-Schlüsselpaar (es können auch andere Algorithmen vom TPM unterstützt werden).
    - Im Allgemeinen darf die Migration zu anderen TPMs erfolgen.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (6/7)

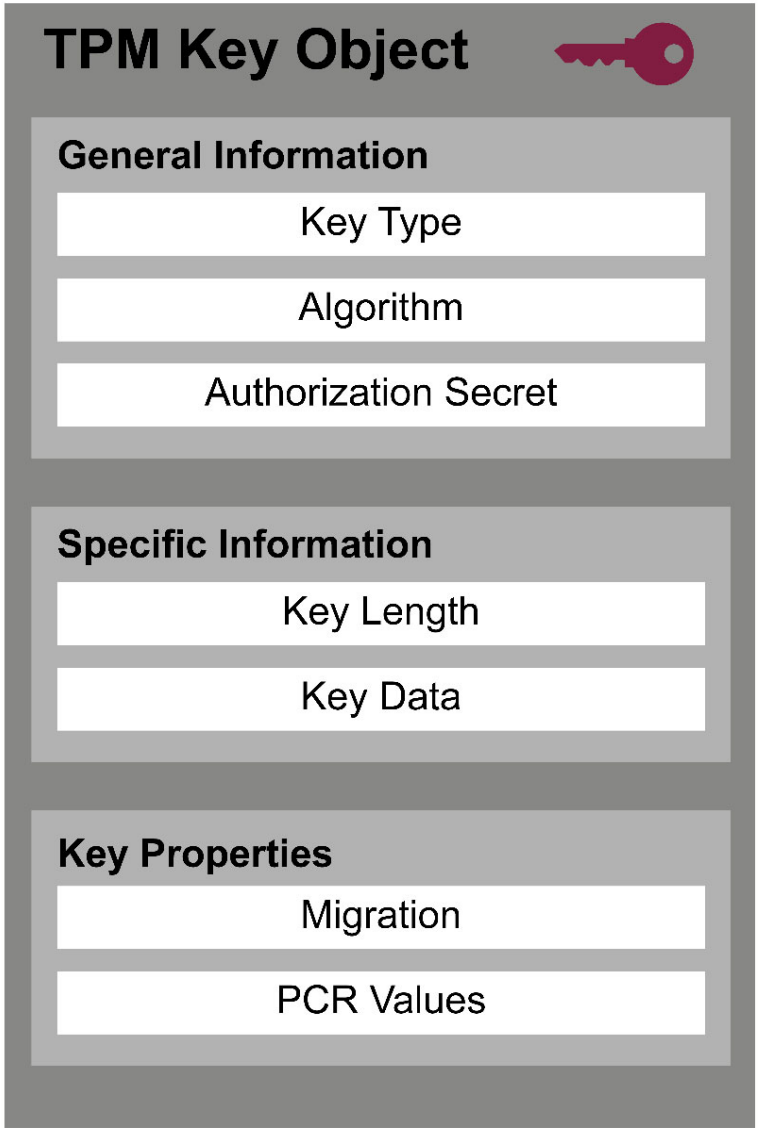
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (6/7)



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

IT-Sicherheitsarchitektur
→ Schlüssel und deren Eigenschaften (7/7)



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- IT-Sicherheitsarchitektur
- **Trusted Computing-Funktion**
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Authenticated Boot /Trusted Boot

- Mithilfe von **Authenticated Boot** (oder Trusted Boot) kann dafür gesorgt werden, dass ein IT-System nur in einem **definierten vertrauenswürdigen Zustand** aktiv wird.

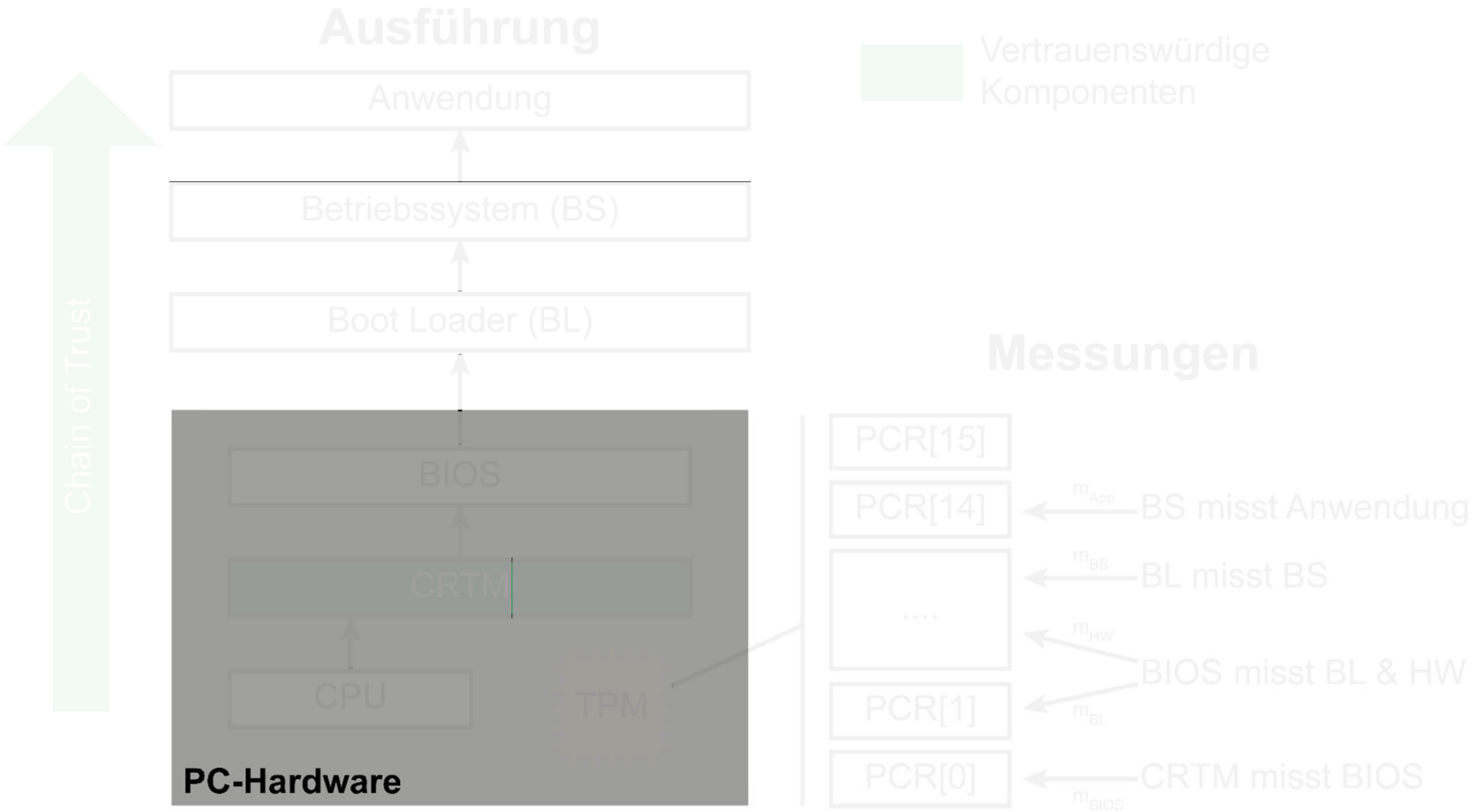
- Dabei werden beim Authenticated Boot die **Systemzustände der Soft- und Hardware** erst mal nur **gemessen** und in die **Platform Configuration Register (PCR) des TPMs gespeichert**.

- **Ablauf:**
  Berechnung des Hashwertes und Schreiben in ein PCR-Register
  - Vertrauenswürdige CRTM Software → BIOS
  - BIOS → Boot Loaders, Hardwarezustand (Mainboard, ROM-Konfigu. …)
  - Boot Loader → Betriebssystems
  - Betriebssystem → Anwendung

- Der **Gesamtwert** kann dann von einer **lokalen Instanz** wie ein Hardware-Sicherheitsmodul eines USB-Sticks oder **aus der Ferne** mit „Remote Attestation“ **überprüft werden**.

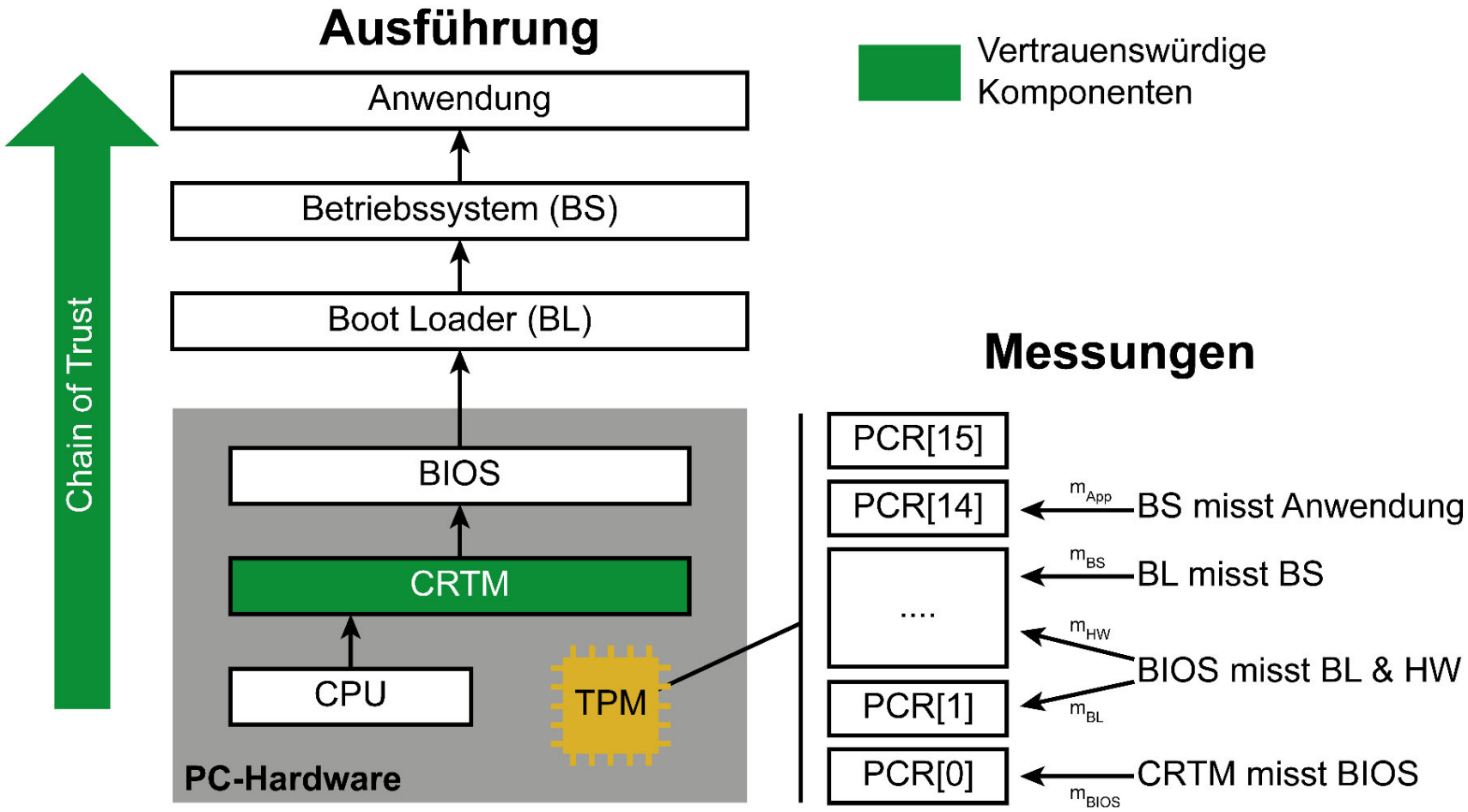
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Sicherheitsarchitektur
→ Authenticated Boot



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Sicherheitsarchitektur
→ Authenticated Boot



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Binding

- Binding wird verwendet, um **Daten** an eine bestimmte **TPM/Plattform** zu **binden**.
- Daten, die mit einem nicht migrierbaren Schlüssel verschlüsselt wurden, können nur von dem entsprechenden TPM wiederhergestellt werden, das den passenden geheimen Schlüssel kennt.
- **Normalerweise** bietet Binding **keine Plattformbindung**, da die Bindung auch mit **migrierbaren Schlüsseln** umgesetzt werden kann.
- Dadurch können **verschlüsselte Daten** auf eine **andere Plattform transferiert** und genutzt werden.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Sealing

- **Sealing bindet Daten** immer an einen bestimmten **TPM** und zusätzlich auch an die **Plattformkonfiguration** (Soft- und Hardwarekonfiguration).
- Sealing kann nur mit **„nicht migrierbaren Schlüsseln“** umgesetzt werden.
- Damit kann die **Soft- und Hardwarekonfiguration** der Plattform verifiziert werden.
- Der **Schlüsseltext enthält implizit** auch den **Status der Plattform** (Soft- und Hardwarekonfiguration) zum Zeitpunkt der Verschlüsselung.
- Dadurch können mit Sealing **Daten an** eine bestimmte **Plattformkonfiguration** gebunden werden.
- Daten können nur entschlüsselt werden, wenn sich die Plattform (Soft- und Hardwarekonfiguration) in einem vordefinierten Zustand befindet.
- Die Idee ist, dass der vordefinierte Zustand ein **vertrauenswürdiger Zustand** ist, mit einer vorgegebenen vertrauenswürdigen Software und Hardware.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Sealing Funktionen und Parameter

Eingabe Parameter

daten	*{unverschlüsselte Daten}*

Ausgabe Parameter

cipher	*{verschlüsselte Daten}*
cryptedKEY	*{verschlüsselter Schlüssel}*

TPM Interne Funktionen und Daten

encrypt (key, daten)	*{symmetrischer Verschlüsselungsalgorithmus „AES“}*
H (daten)	*{One-Way-Hashfunktion “SHA-256”}*
genKey()	*{Schlüsselerzeugung}*
SRK	*{Storage Root Key}*
PCRs	*{PCR-0, PCR-1, …} z.B. aktuell abgespeicherte PCR-Werte*

plainKEY = genKEY ()

cipher = encrypt (plainKEY, (daten // H (daten // PCR-0 // … // PCR-x)))

cryptedKEY = encrypt (SRK, plainKEY // H (plainKEY))

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Sealing Funktionen und Parameter

Eingabe Parameter

daten	*{unverschlüsselte Daten}*

Ausgabe Parameter

cipher	*{verschlüsselte Daten}*
cryptedKEY	*{verschlüsselter Schlüssel}*

TPM Interne Funktionen und Daten

encrypt (key, daten)	*{symmetrischer Verschlüsselungsalgorithmus „AES“}*
H (daten)	*{One-Way-Hashfunktion “SHA-256”}*
genKey()	*{Schlüsselerzeugung}*
SRK	*{Storage Root Key}*
PCRs	*{PCR-0, PCR-1, …} z.B. aktuell abgespeicherte PCR-Werte*

plainKEY = genKEY ()

cipher = encrypt (plainKEY, (daten // H (daten // PCR-0 // … // PCR-x))

cryptedKEY = encrypt (SRK, plainKEY // H (plainKEY))

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Un-Sealing Funktionen und Parameter

Eingabe Parameter

cipher	*{verschlüsselte Daten}*
cryptedKEY	*{verschlüsselter Schlüssel}*

Ausgabe Parameter

daten	*{unverschlüsselte Daten}*

TPM Interne Funktionen und Daten

decrypt (key, daten)	*{symmetrischer Verschlüsselungsalgorithmus „AES“}*
H (daten)	*{One-Way-Hashfunktion “SHA-256”}*
checkPCRs (Hash-Value)	*{vergleicht PCRs-Inhalte mit Hash-Value}*
SRK	*{Storage Root Key}*
PCRs	*{PCR-0, PCR-1, …}*

plainKEY = decrypt (SRK, cryptedKEY)

daten // H (daten // PCR-0 // … // PCR-x) = decrypt (plainKEY, cipher)

```
if ( checkPCRs ( Hash-Value ) )
        return daten
else
        return ERROR
```

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Un-Sealing Funktionen und Parameter

Eingabe Parameter

cipher	*{verschlüsselte Daten}*
cryptedKEY	*{verschlüsselter Schlüssel}*

Ausgabe Parameter

daten	*{unverschlüsselte Daten}*

TPM Interne Funktionen und Daten

decrypt (key, daten)	*{symmetrischer Verschlüsselungsalgorithmus „AES“}*
H (daten)	*{One-Way-Hashfunktion “SHA-256”}*
checkPCRs (Hash-Value)	*{vergleicht PCRs-Inhalte mit Hash-Value}*
SRK	*{Storage Root Key}*
PCRs	*{PCR-0, PCR-1, …}*

plainKEY = decrypt (SRK, cryptedKEY)

daten // H (daten // PCR-0 // … // PCR-x) = decrypt (plainKEY, cipher)

```
if ( checkPCRs ( Hash-Value ) )
        return daten
else
        return ERROR
```

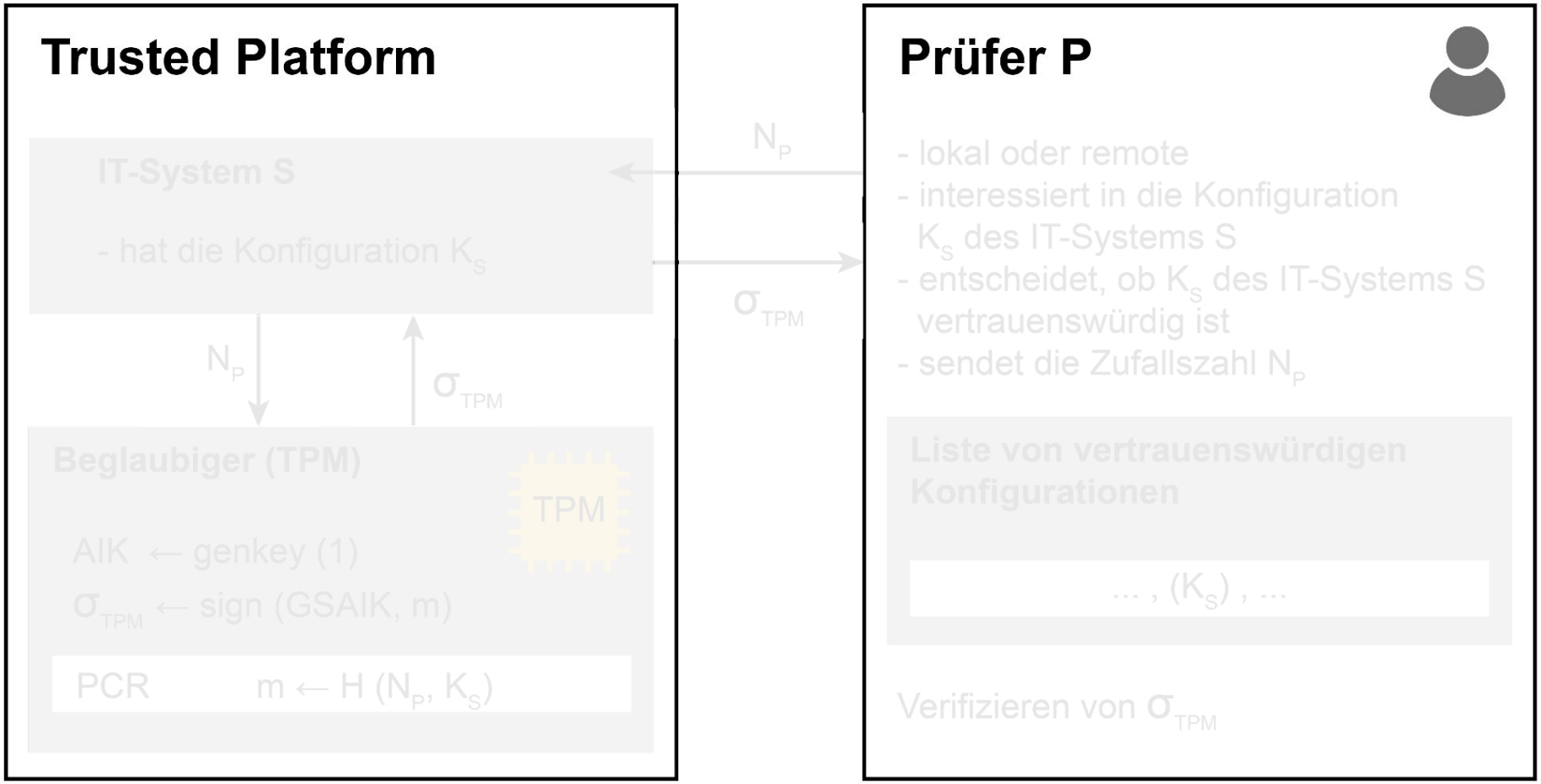
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Remote Attestation

- Mit (Remote) **Attestation** ist es möglich, die aktuelle Konfiguration eines IT-Systems zu überprüfen.
- Eine prüfende Instanz kann lokal oder remote überprüfen, ob eine gewünschte vertrauenswürdige Konfiguration eines IT-Systems vorliegt.
- Dazu hat die prüfende Instanz eine Liste von vertrauenswürdigen Konfigurationen, die ein IT-System haben kann, damit es gestartet werden kann oder eine Interaktion umgesetzt wird.
- **Remote Attestation** gibt die Möglichkeit, die **Vertrauenswürdigkeit** von **anderen, auch fremden IT-Systemen zu messen**, bevor eine Interaktion mit diesem IT-System begonnen wird.

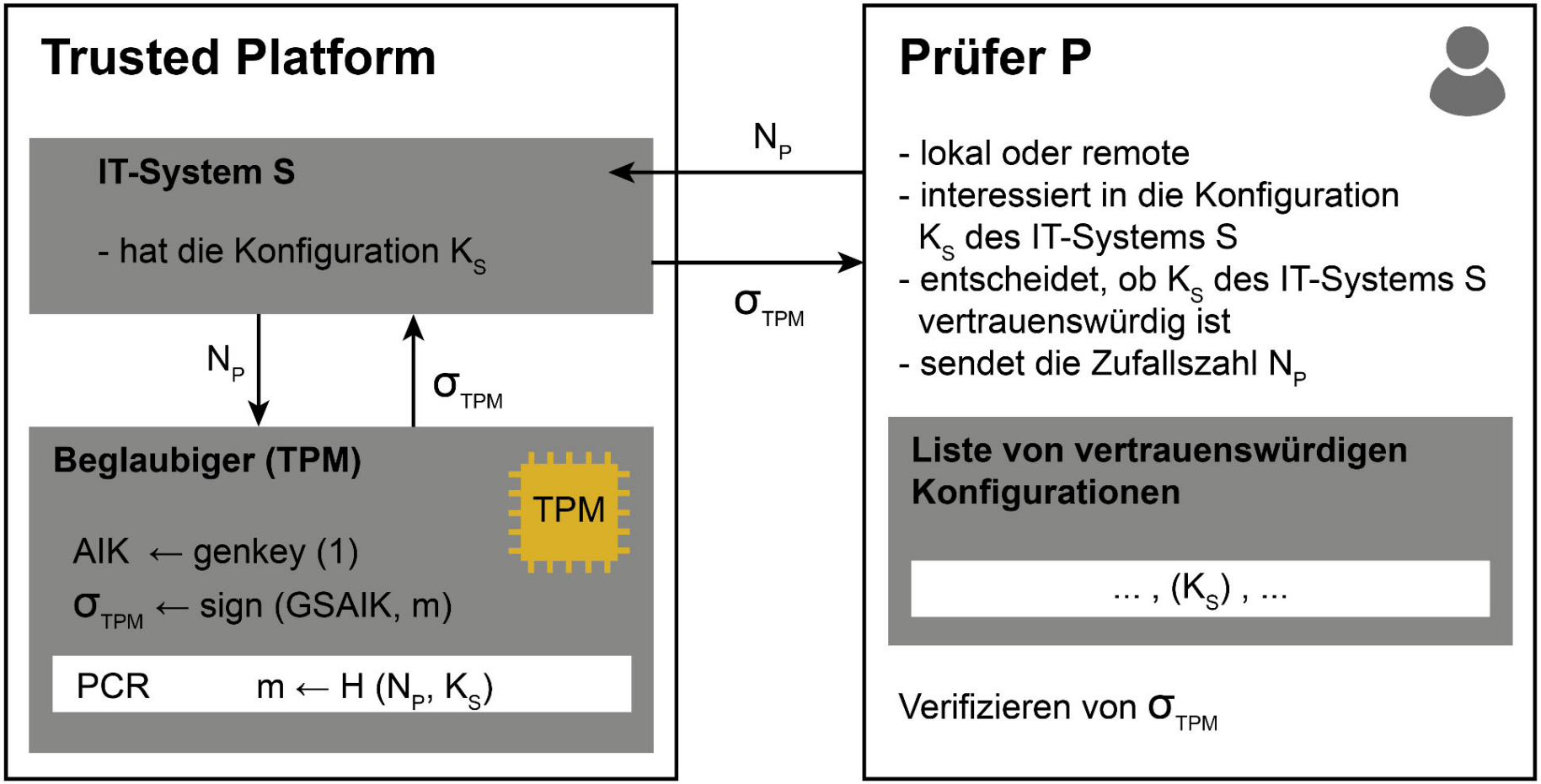
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ (Remote) Attestation Ablauf



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ (Remote) Attestation Ablauf



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Signaturfunktion für die Attestierung

Eingabe Parameter

random	*{Zufallszahl des Prüfers P – N_P}*

Ausgabe Parameter

signature//certificate	*{Signatur der aktuellen Systemkonfiguration des AIKs}*

TPM Interne Funktionen und Daten

sign (key, daten)	*{RSA-Signatur}*
H (daten)	*{One-Way-Hashfunktion "SHA-256"}*
GSAIK	*{geheimer AIK-RSA-Schlüssel}*
AIK-certificate	*{elektronisches Zertifikat des AIKs}*
PCRs	*{PCR-0, PCR-1, …} z.B. aktuell abgespeicherte PCR-Werte*

σ_{TPM} = *sign (GSAIK, H (random // PCR-0 // … // PCR-x))*

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Signaturfunktion für die Attestierung

Eingabe Parameter

random	*{Zufallszahl des Prüfers P – N_P}*

Ausgabe Parameter

signature//certificate	*{Signatur der aktuellen Systemkonfiguration des AIKs}*

TPM Interne Funktionen und Daten

sign (key, daten)	*{RSA-Signatur}*
H (daten)	*{One-Way-Hashfunktion "SHA-256"}*
GSAIK	*{geheimer AIK-RSA-Schlüssel}*
AIK-certificate	*{elektronisches Zertifikat des AIKs}*
PCRs	*{PCR-0, PCR-1, …} z.B. aktuell abgespeicherte PCR-Werte*

$$\sigma_{TPM} = sign\ (\ GSAIK,\ H\ (\ random\ //PCR\text{-}0\ //\cdots\ //PCR\text{-}x\ )\ )$$

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Verifikationsfunktion für die Attestierung

Eingabe Parameter

signature//certificate	*{Signatur der Systemkonfiguration des AIKs}*

Ausgabe Parameter

return value	*{Rückgabewert}*

TPM Interne Funktionen und Daten

very (key, daten)	*{RSA-Signatur-Verifikation}*
H (daten)	*{One-Way-Hashfunktion "SHA-256"}*
ÖSAIK	*{öffentlicher AIK-RSA-Schlüssel}*
checkPCRs (PCR-Values)	*{vergleicht den Inhalt der PCRs mit den gewünschten Werten}*
PCRs	*{PCR-0, PCR-1, ···}*

```
if ( very ( ÖSAIK, σTPM ) )  and
if ( checkCERT ( certificate ) )      and
if ( checkPCRs ( Hash-Value ) )
        return ok
else
        return ERROR
```

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing-Funktion
→ Verifikationsfunktion für die Attestierung

Eingabe Parameter

signature//certificate	*{Signatur der Systemkonfiguration des AIKs}*

Ausgabe Parameter

return value	*{Rückgabewert}*

TPM Interne Funktionen und Daten

very (key, daten)	*{RSA-Signatur-Verifikation}*
H (daten)	*{One-Way-Hashfunktion "SHA-256"}*
ÖSAIK	*{öffentlicher AIK-RSA-Schlüssel}*
checkPCRs (PCR-Values)	*{vergleicht den Inhalt der PCRs mit den gewünschten Werten}*
PCRs	*{PCR-0, PCR-1, …}*

```
if ( very ( ÖSAIK, σTPM ) )  and
if ( checkCERT ( certificate ) )      and
if ( checkPCRs ( Hash-Value ) )
        return ok
else
        return ERROR
```

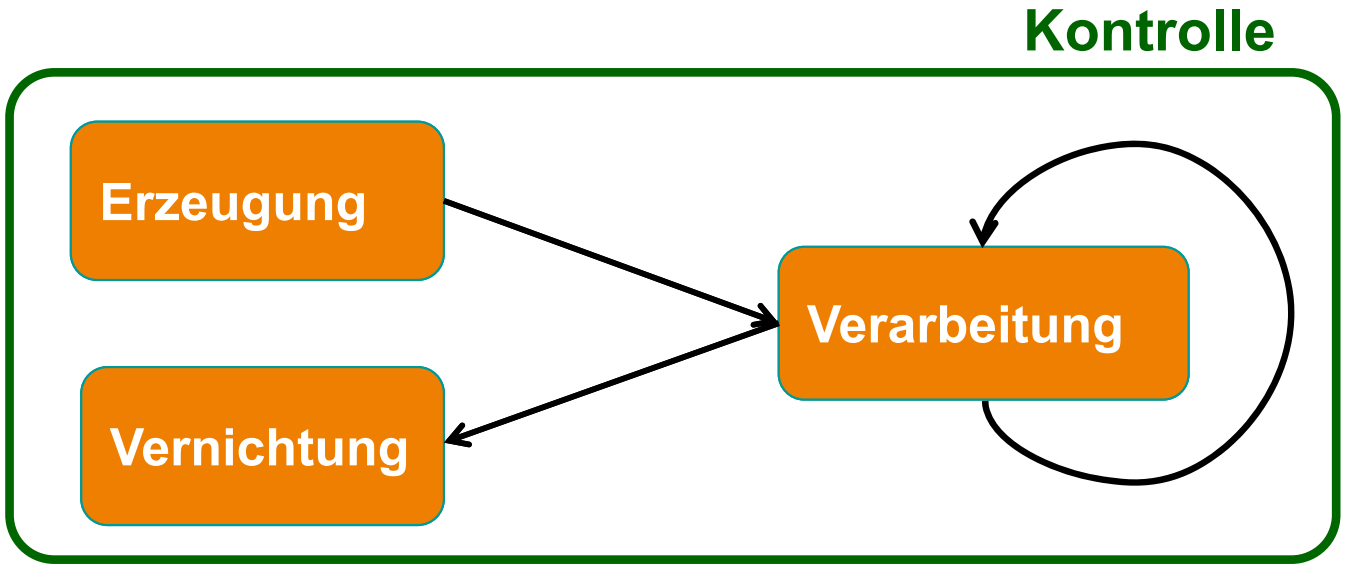
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Objekt-Sicherheit
→ Remote Attestation

Idee: Domänenorientierte Objektsicherheit, bei der die Objekte mit Rechten versehen werden, die definieren, wer sie in welcher IT-Umgebung wie nutzen darf.

→ *Object Lifecycle Protection*

→ *Distributed Policy Enforcement (**even on foreigen systems**)*



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- **Trusted Platform - Turaya**
- Trusted Network Connect - TNC
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Übersicht

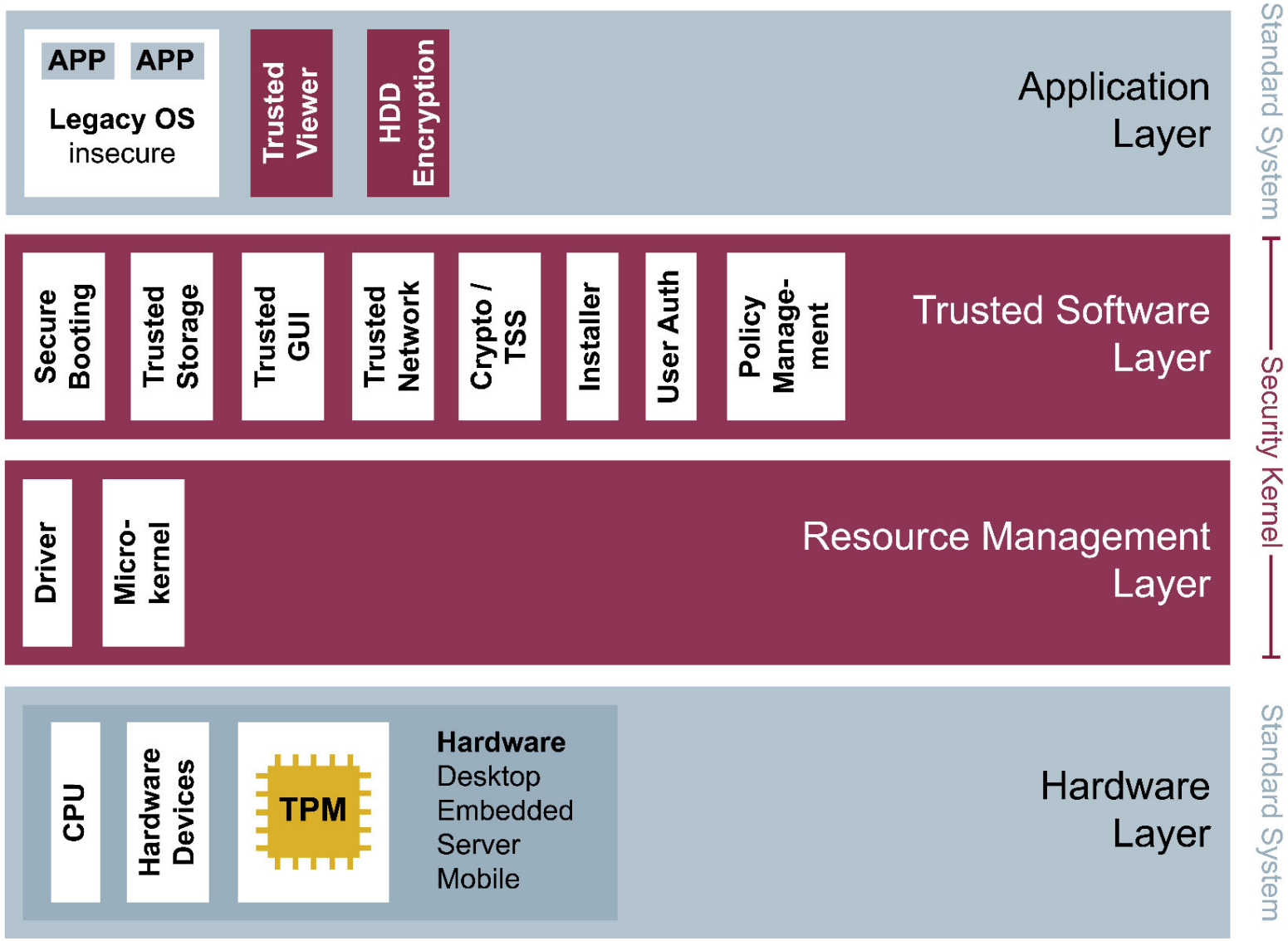
- Eine **Trusted Platform** (IT-Sicherheitsplattform) **löst** mithilfe von **Trusted Computing-Technologien** essenzielle **IT-Sicherheits-Probleme** konventioneller IT-Systeme, wie sie durch SW-Schwachstellen und Malware verursacht werden.
- Darüber hinaus ermöglicht die IT-Sicherheitsplattform die **Durchsetzung eigener Sicherheitsregeln** bei der Ausführung eigener Inhalte und Dokumente **auf fremden IT-Systemen**.
- Im Folgenden werden die technische Architektur der **IT-Sicherheitsplattform Turaya** skizziert und die grundlegenden Eigenschaften dargelegt.
- Außerdem werden Beispielanwendungen beschrieben.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Übersicht

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Übersicht

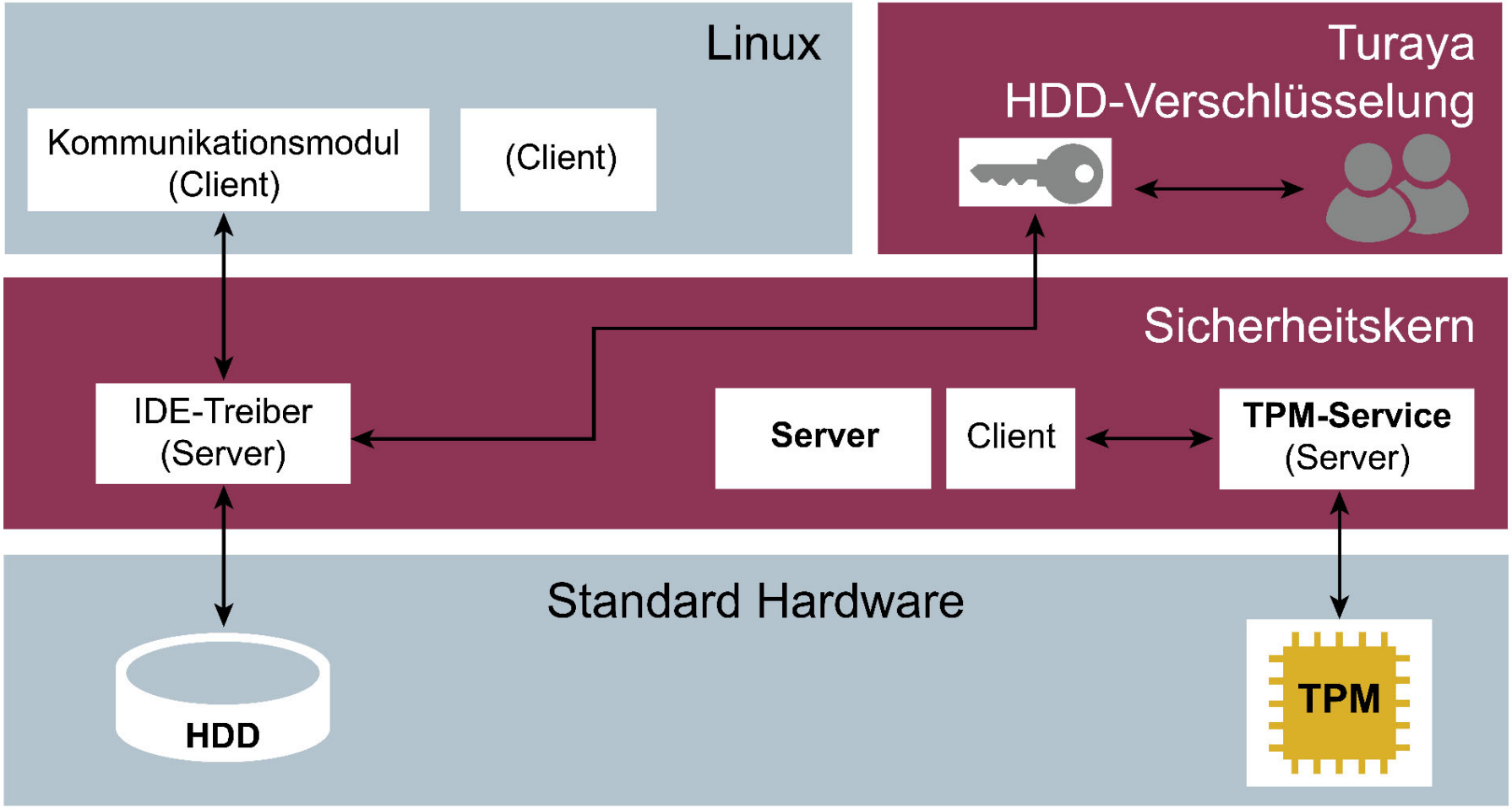


© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.Crypt

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.Crypt

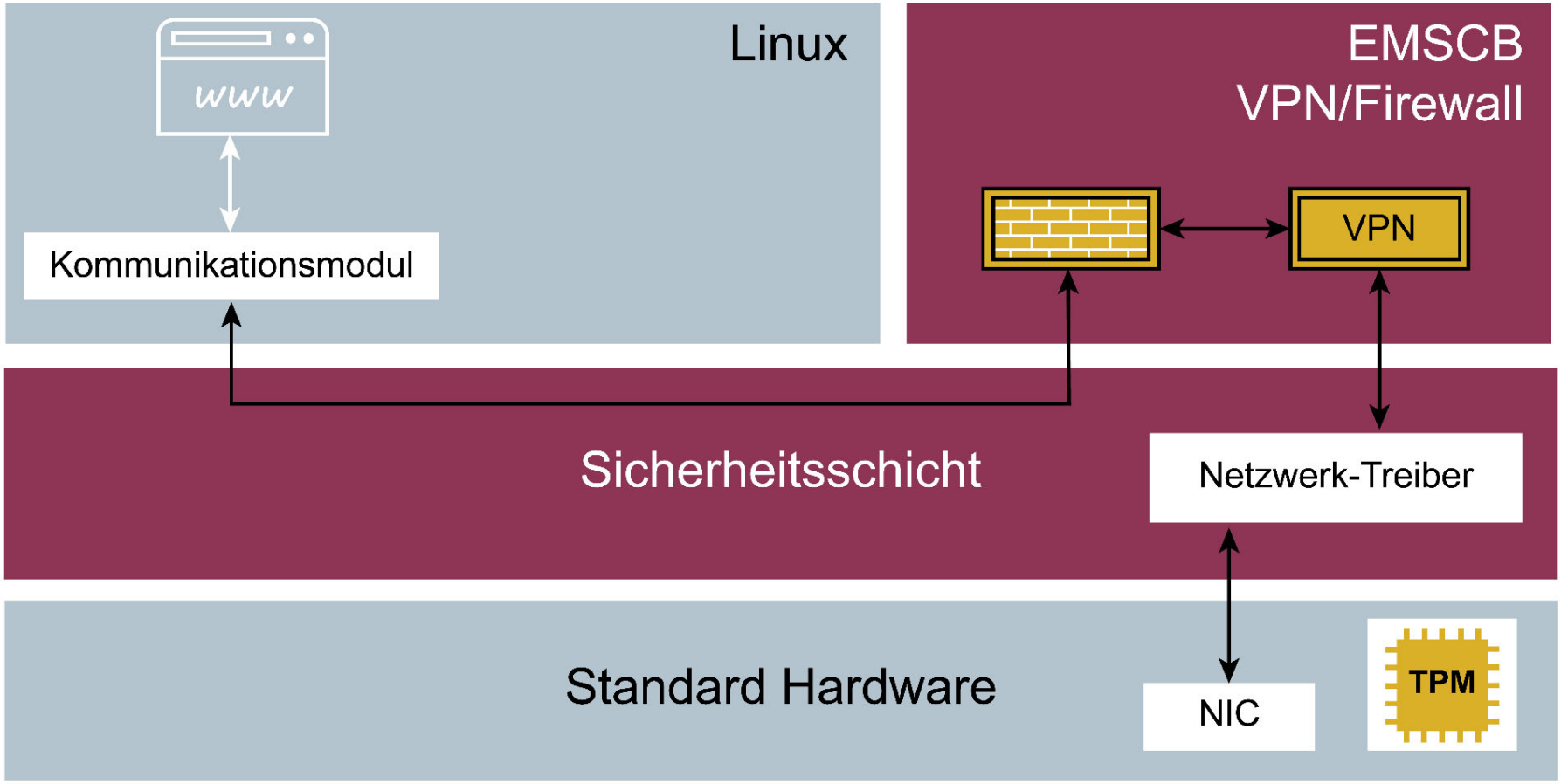


© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.VPN

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.VPN

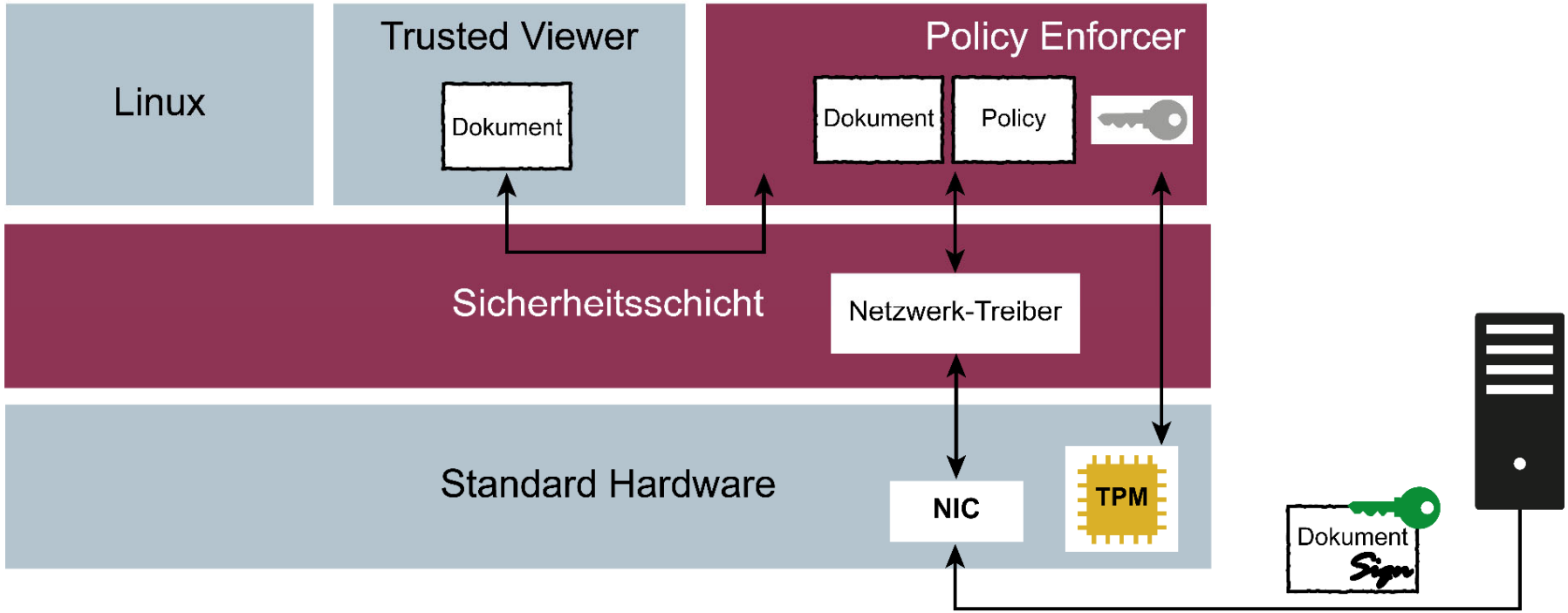


© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.FairDRM

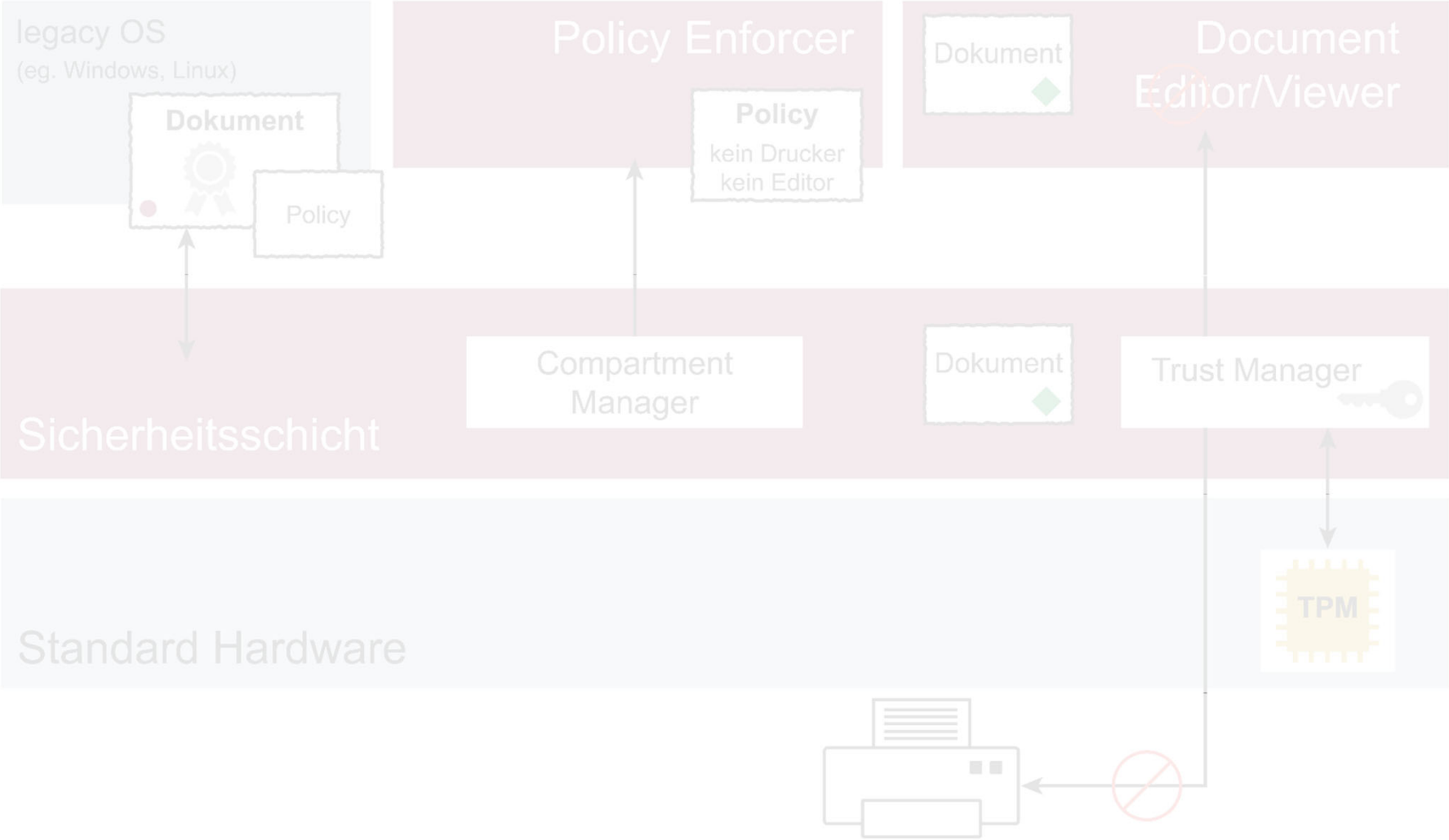
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.FairDRM



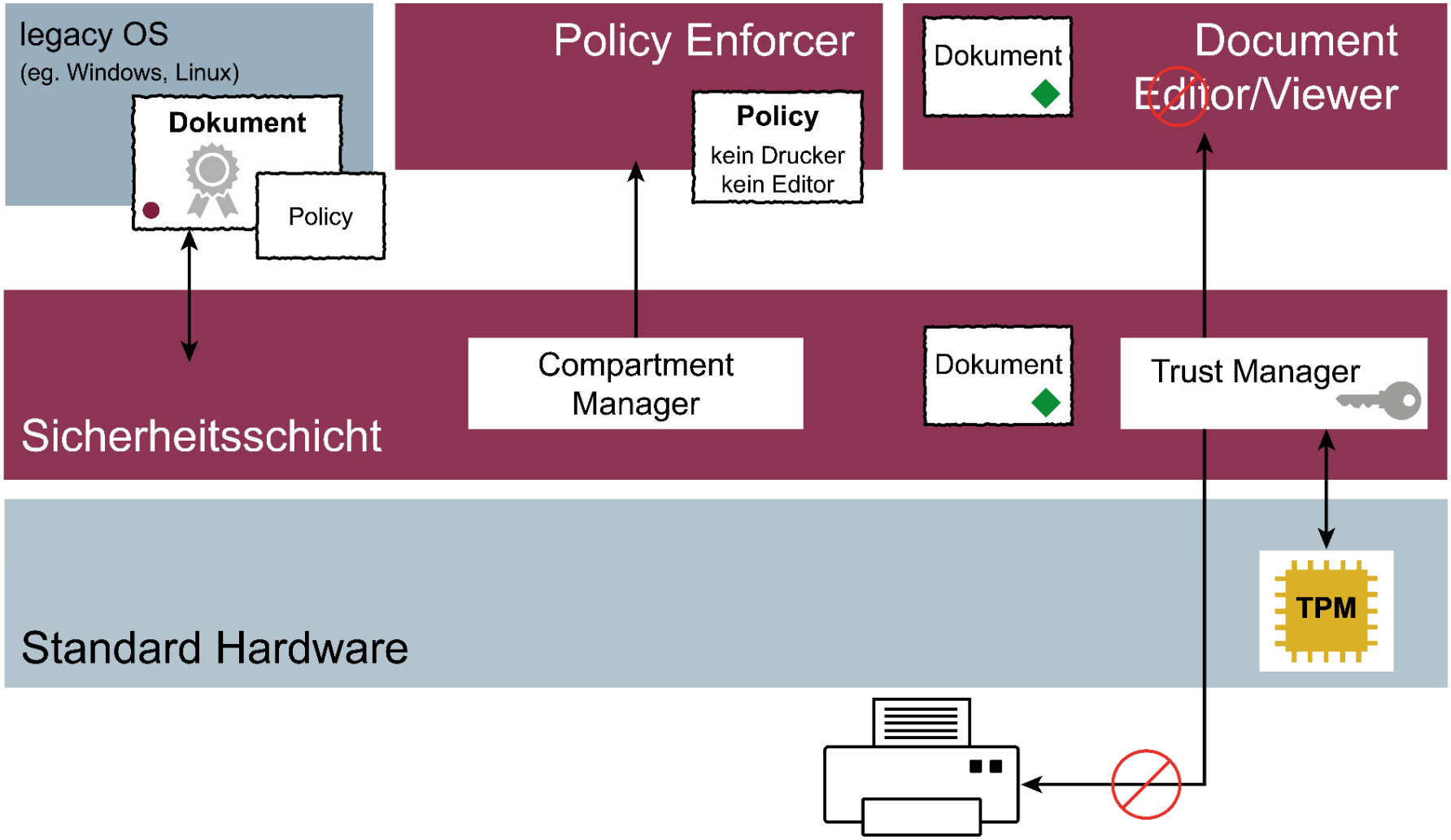
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.ERM



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Platform - Turaya
→ Beispielanwendung: Turaya.ERM



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

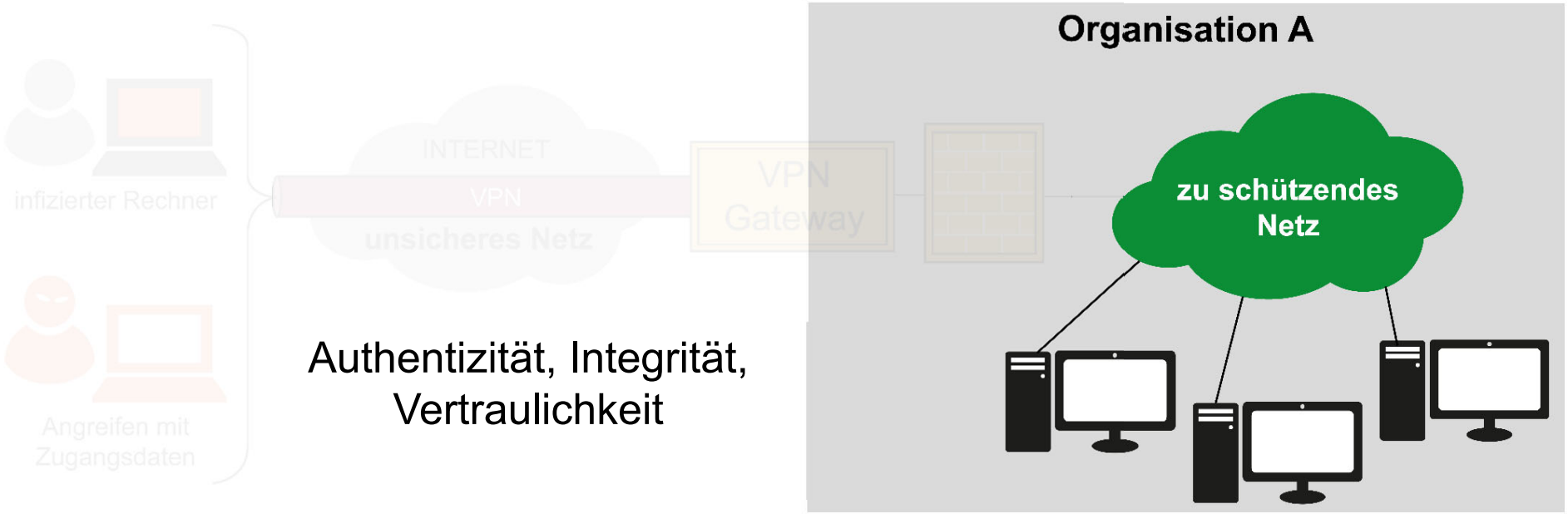
Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- **Trusted Network Connect - TNC**
- Zusammenfassung

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Problemstellung

Vertrauen?



Authentizität, Integrität, Vertraulichkeit

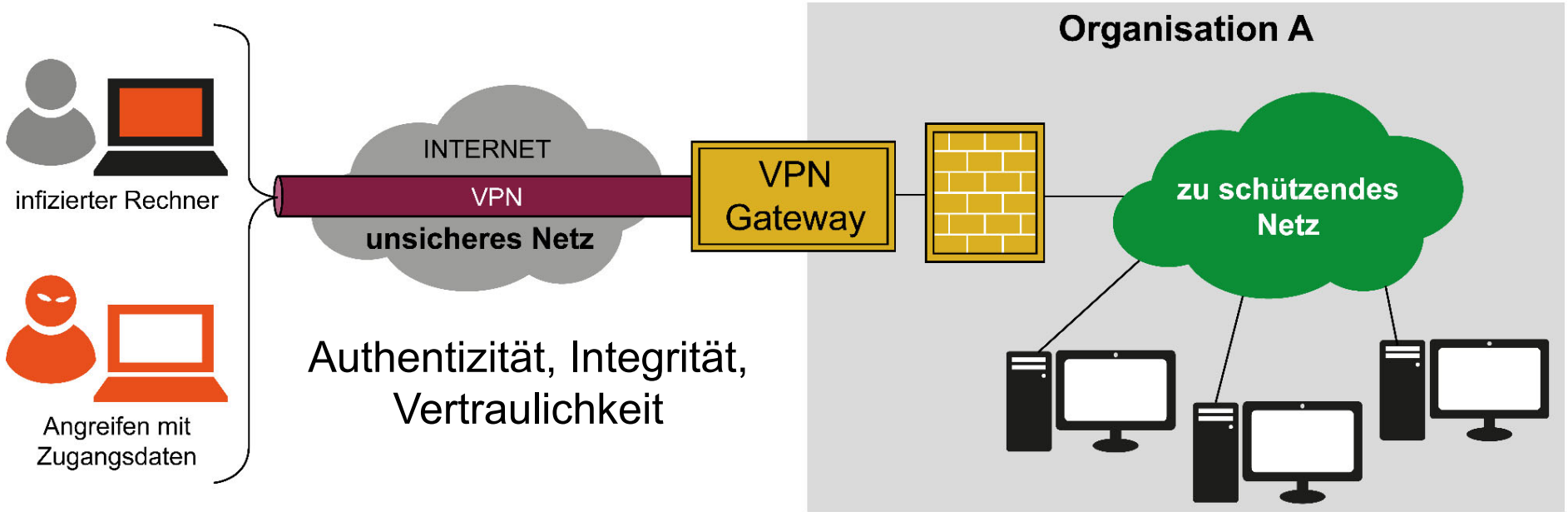
Lösungsansätze:

→ Microsoft NAP
→ Cisco NAC
→ Trusted Computing Group (TCG) → **Trusted Network Connect (TNC)**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Problemstellung

Vertrauen?



Lösungsansätze:

→ Microsoft NAP

→ Cisco NAC

→ Trusted Computing Group (TCG) → **Trusted Network Connect (TNC)**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

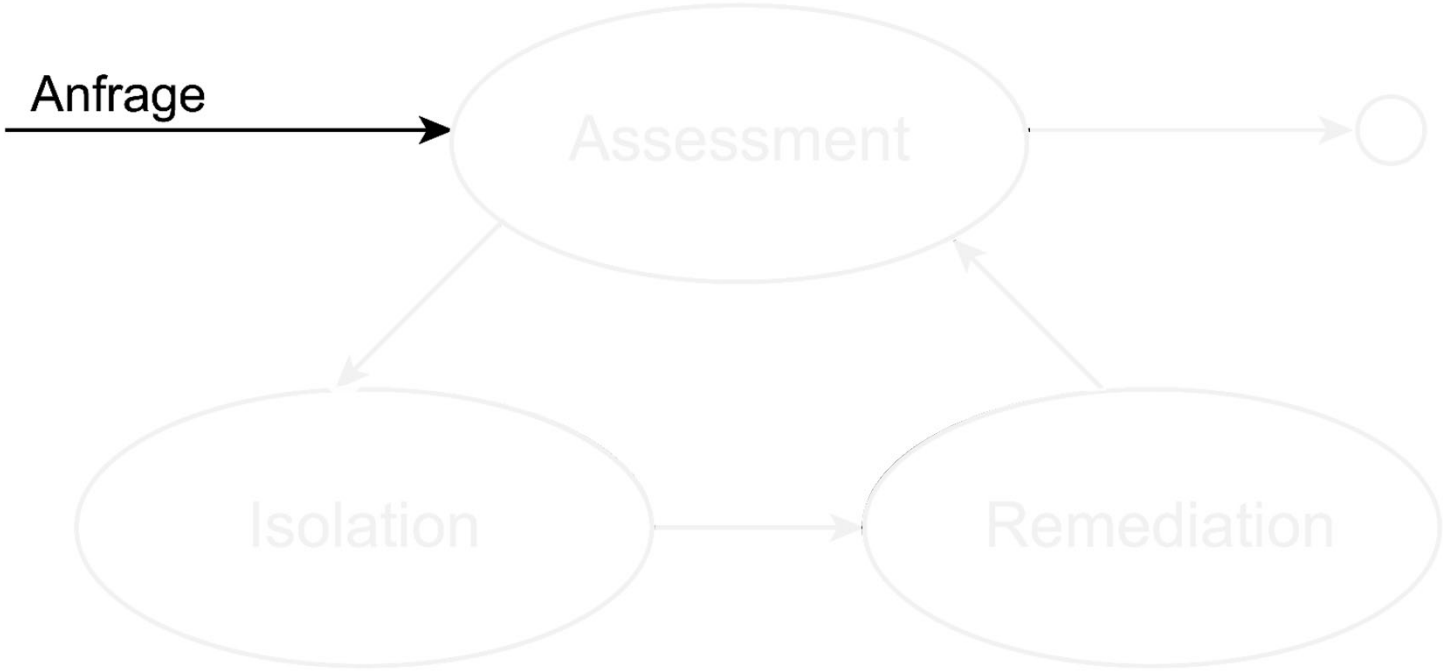
Trusted Network Connect

→ Vertrauenswürdige Netzwerkverbindungen

- Vertrauenswürdigkeit abhängig von der **Integrität**.
- **Alle** beteiligten **Kommunikationspartner** betrachten.
- **Kommunikationsrichtung** getrennt betrachten.
- Alle IT-Systeme, Infrastrukturelemente und das Umfeld der Kommunikation bewerten.
- Anforderungen in **Policies** definieren (z.B. erlaubte Konfigurationen, Betriebssysteme, Hard- und Software, …).
- Überprüfung **vor dem** Zugriff (**Prävention**).

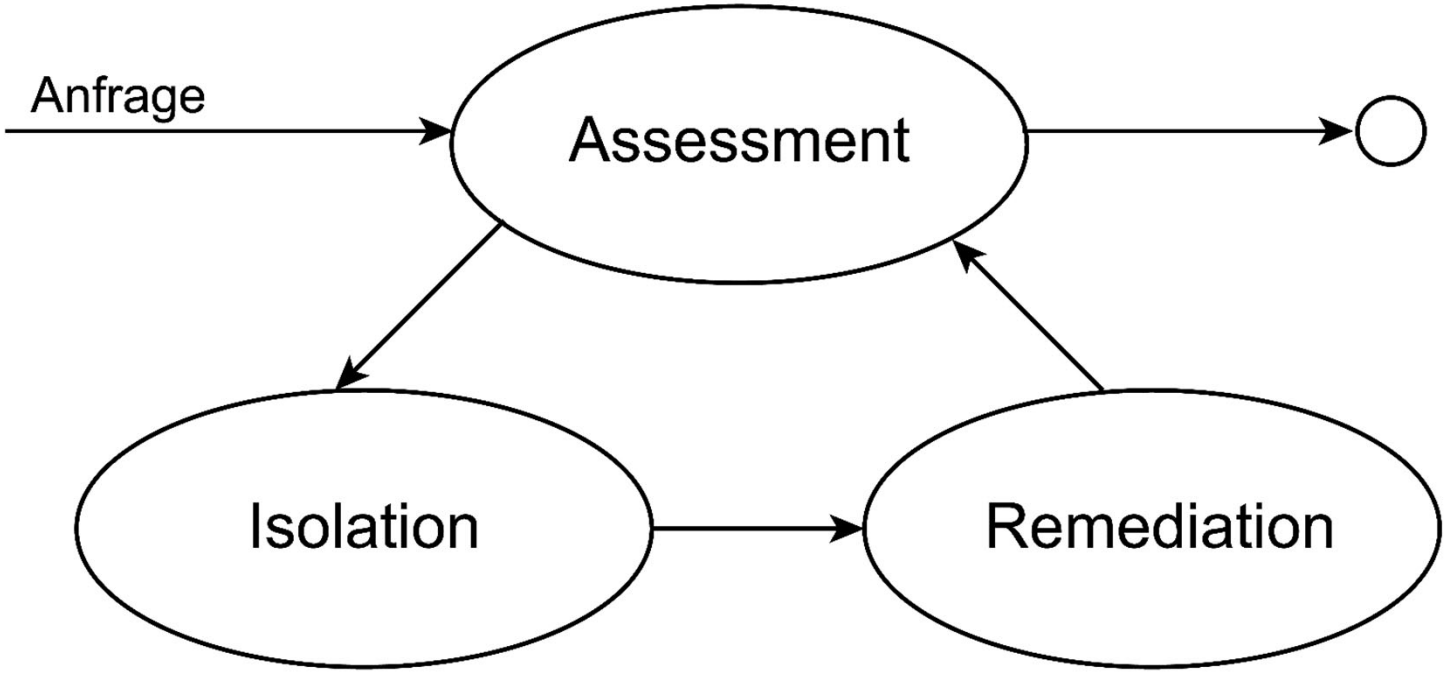
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Phasen



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Phasen



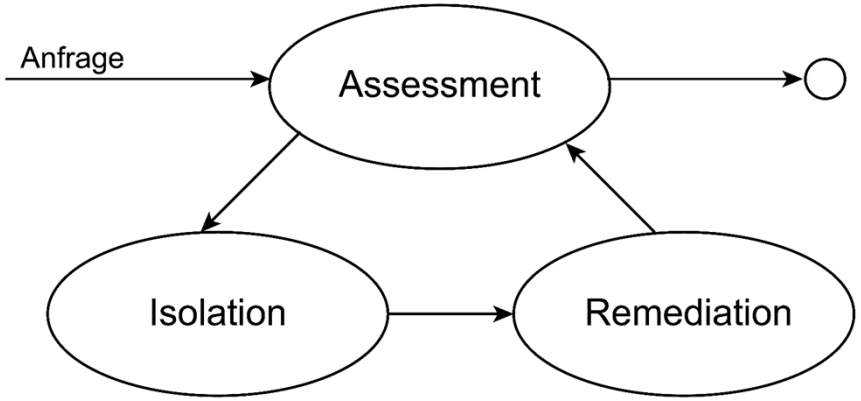
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Assessment-Phase

- Die Assessment-Phase umfasst **alle Aktionen** vom **Versuch eines Verbindungsaufbaus** zu einem TNC-geschützten Netzwerk bis zur **Entscheidung über** dessen **Vertrauenswürdigkeit**.

- In dieser Phase werden **Messwerte vom IT-System** an einen Server im Netzwerk gesendet und dort anhand von Policys verglichen.

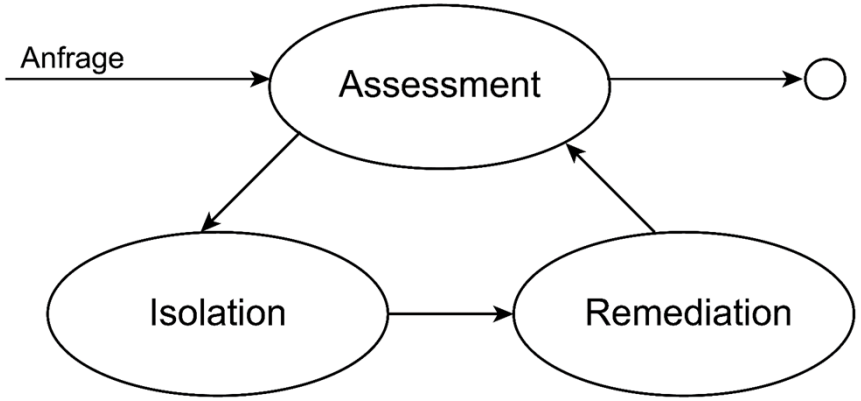
- Durch diesen Vergleich ist eine **Entscheidung über die Vertrauenswürdigkeit** möglich.



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Isolation-Phase

- Wird das IT-System, bei **Nichterfüllung der Policys**, als nicht-vertrauenswürdig eingestuft, gelangt es in die **Isolation-Phase**.
- In dieser Phase wird das zugreifende IT-System in einen geschützten Netzwerkbereich isoliert, der vom restlichen Netz abgeschottet ist.
- Eventuell mit Malware kompromittierte IT-Systeme oder IT-Systeme von Angreifern **erlangen** so **keinen Zugriff auf das Netzwerk** und die dort angebotenen Dienste.

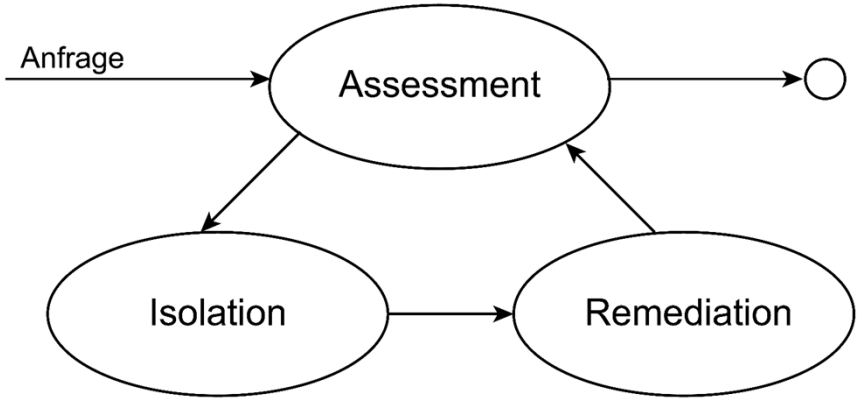


© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Remediation-Phase

- Die Remediation-Phase bietet den isolierten IT-Systemen die Möglichkeit, ihre **Integrität**, zum Beispiel über die Installation fehlender Sicherheitssoftware, **wiederherzustellen**, und nach einer erneuten Überprüfung Zugriff auf das Netzwerk mit seinen angebotenen Dienste zu erlangen.

- Sowohl die Isolation- als auch die Remediation-Phase sind durch die TNC-Spezifikation nicht vorgeschrieben und müssen somit nicht zwangsläufig implementiert werden.



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ IT-System mit dem Access Requestor

- Das **IT-System**, mit dem eine Netzwerkverbindung zu einem TNC-Netzwerk aufgebaut werden soll, wird **Access Requestor (AR)** genannt.
- Auf dem AR befinden sich TNC-Komponenten für Verbindungsanfrage, Messwertermittlung und Messwertübermittlung.
- Die Messung der einzelnen Komponenten des IT-Systems findet durch sogenannte **„Integrity Measurement Collectors“** (IMC) statt.
- Für jede zu messende sicherheitsrelevante Komponente oder Cyber-Sicherheitslösung existiert dabei ein passender IMC, z.B. einer für das installierte Betriebssystem und SW-Anwendungen, für die installierte Hardware, Anti-Malwarelösung und einer für die Personal-Firewall.
- Zum Systemstart werden die IMCs vom TNC-Client auf dem zugreifenden IT-System initialisiert, um bei einem Verbindungsaufbau Messwerte von den jeweiligen Komponenten sammeln zu können.
- Die Art der möglichen Messwerte ist dabei zunächst nicht begrenzt.
- Bei einer Anti-Malware-Lösung können z.B. Informationen über Hersteller und Alter der Malware-Signatur wichtig sein, bei einem angeschlossenen Drucker die Version der Firmware.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Policy Decision Point (PDP) – 1/2

- Der Policy Decision Point (PDP) stellt die Gegenseite zum Access Requestor (AR) dar.
- Es handelt sich dabei um einen Server, der die **Aufgabe** hat, die **Messwerte eines Access Requestors zu sammeln** und mithilfe von Policys eine **Zugriffsentscheidung** zu **formulieren**.
- Diese Entscheidung wird anschließend der ausführenden Stelle für den Zugriff mitgeteilt.
- Die Network Access Authority (NAA) im Policy Decision Point entscheidet, ob ein AR Zugriff bekommen soll oder nicht.
- Dazu fragt der NAA den TNC-Server, ob die Integritätsmessungen des ARs mit der Security Policy übereinstimmen.
- Auf dem PDP stellen sogenannte „Integrity Measurement Verifier" (IMV) das Gegenstück zu den IMCs des AR dar.
- Auch hier existieren mehrere IMVs für die unterschiedlichen Aspekte und Cyber-Sicherheitskomponenten und Cyber-Sicherheitslösungen.
- Für jeden zu überprüfenden Aspekt und jede Cyber-Sicherheitskomponente muss es, neben dem IMC, auch einen passenden IMV geben.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Policy Decision Point (PDP) –2/2

- Damit ein IMC die richtigen Messwerte sammeln kann, bedarf es einer sehr genauen Kenntnis der zu messenden Hardware/Software.
- Diese Kenntnis besitzt meist nur der Hersteller von Hardware beziehungsweise Software.
- Daher ist eine direkte Verbindung zum Hersteller eines IMC sehr hilfreich.
- Die IMVs vergleichen die übermittelten Messwerte anhand der in den Policys festgelegten Regeln und teilen ihr Ergebnis dem TNC-Server im PDP mit.
- Dieser kann mit diesen Teilergebnissen eine Entscheidung über den Zugriff auf das Netzwerk mit seinen Diensten treffen und diese Gesamtentscheidung dem Policy Enforcement Point über die Network Access Authority (NAA) mitteilen.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Policy Enforcement Point (PEP) – 1/2

- Der Policy Enforcement Point (PEP) ist das TNC-Element am Eintrittspunkt des Netzes.
- Seine **Aufgaben** sind die Entgegennahme und Weiterleitung von Verbindungsanfragen sowie die **Ausführung der Handlungsentscheidung** des PDPs.
- Der PEP stellt als Eintrittspunkt den zuerst zu adressierenden Verbindungspunkt des Netzwerkes dar.
- Ankommende Verbindungsanfragen eines AR werden direkt an den PDP weitergeleitet.
- Nachdem ein PDP seine Entscheidung über die Vertrauenswürdigkeit des AR getroffen hat, teilt er diese dem PDP mit, der gemäß dieser Entscheidung handeln muss.

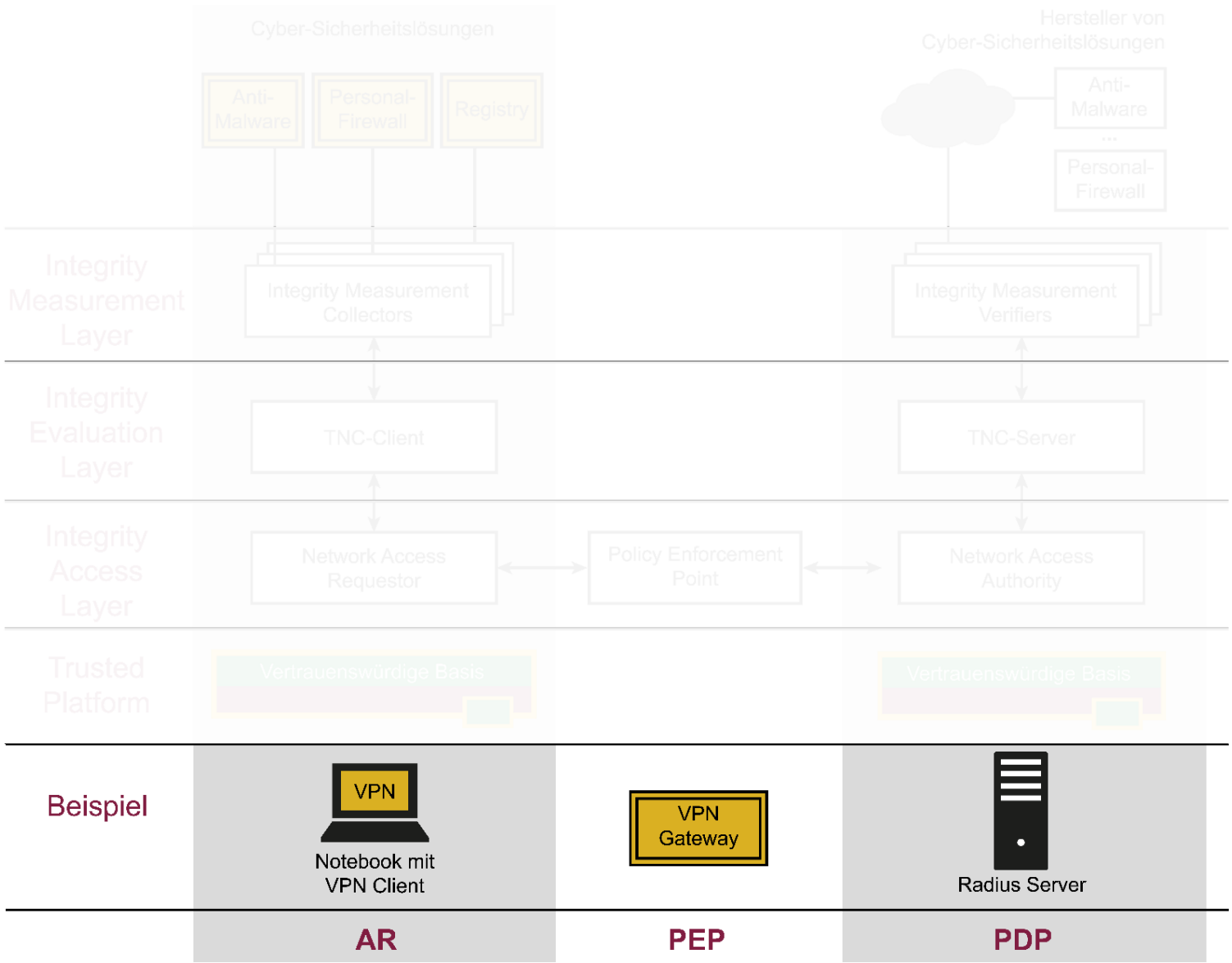
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Policy Enforcement Point (PEP) – 2/2

- Ein PEP kann laut TNC-Spezifikation sowohl ein eigenständiges IT-System als auch in den PDP oder in ein anderes Netzwerk-Equipment integriert sein.
- Dadurch wird es möglich, den PEP zum Beispiel flexibel direkt in ein VPN-Gateway zu integrieren oder, um vorhandene Netzwerkstrukturen unberührt zu lassen, vor beziehungsweise hinter diesem Gateway zu platzieren.
- Wenn AR und PDP auf einer **vertrauenswürdigen Basis** arbeiten, kann den gemessenen Messwerten eine höhere Echtheit zugesprochen werden, weil diese nicht manipuliert werden können.
- Damit wird der **Lying Endpoint-Problematik** entgegengewirkt.

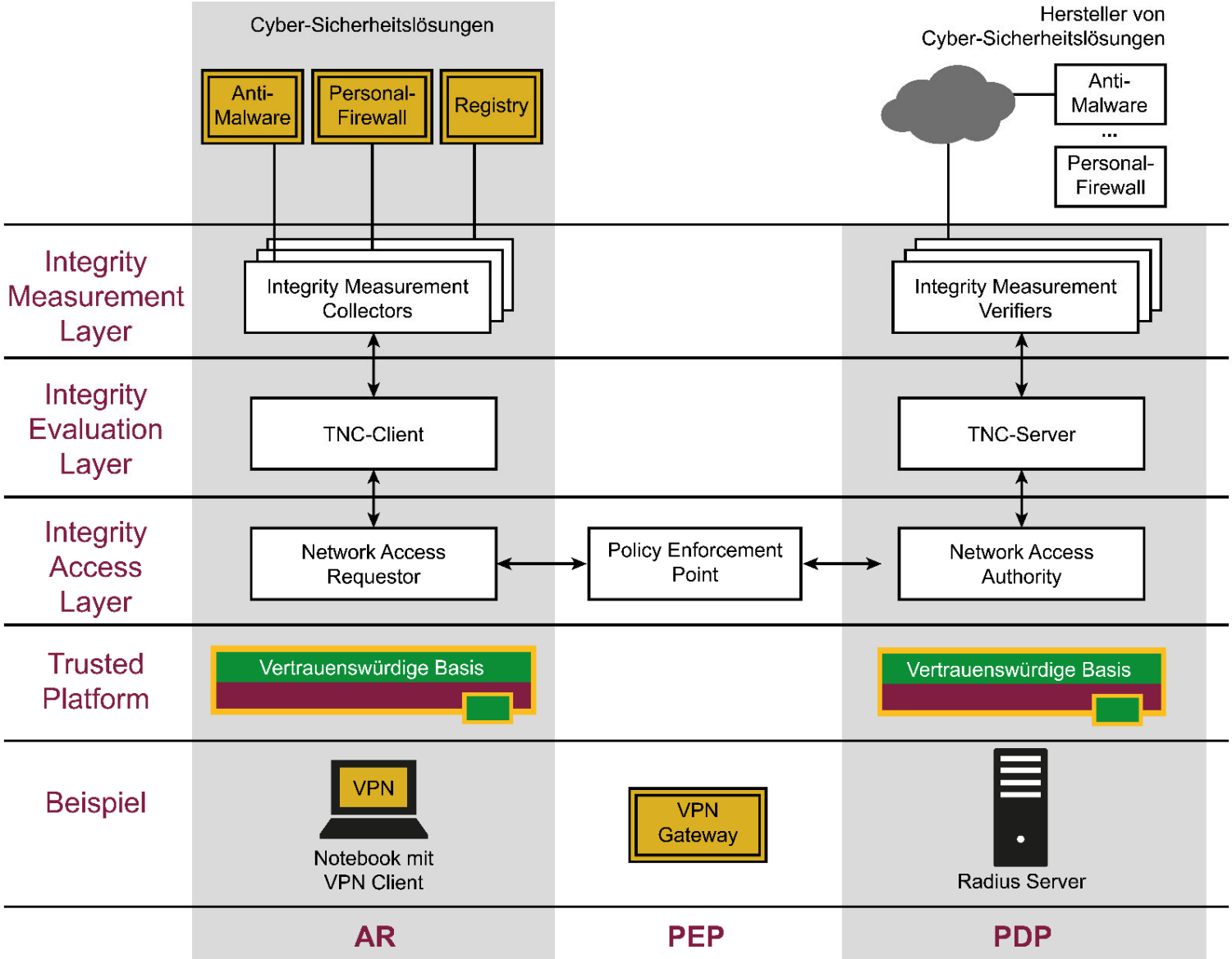
© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Network Connect
→ Struktur



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

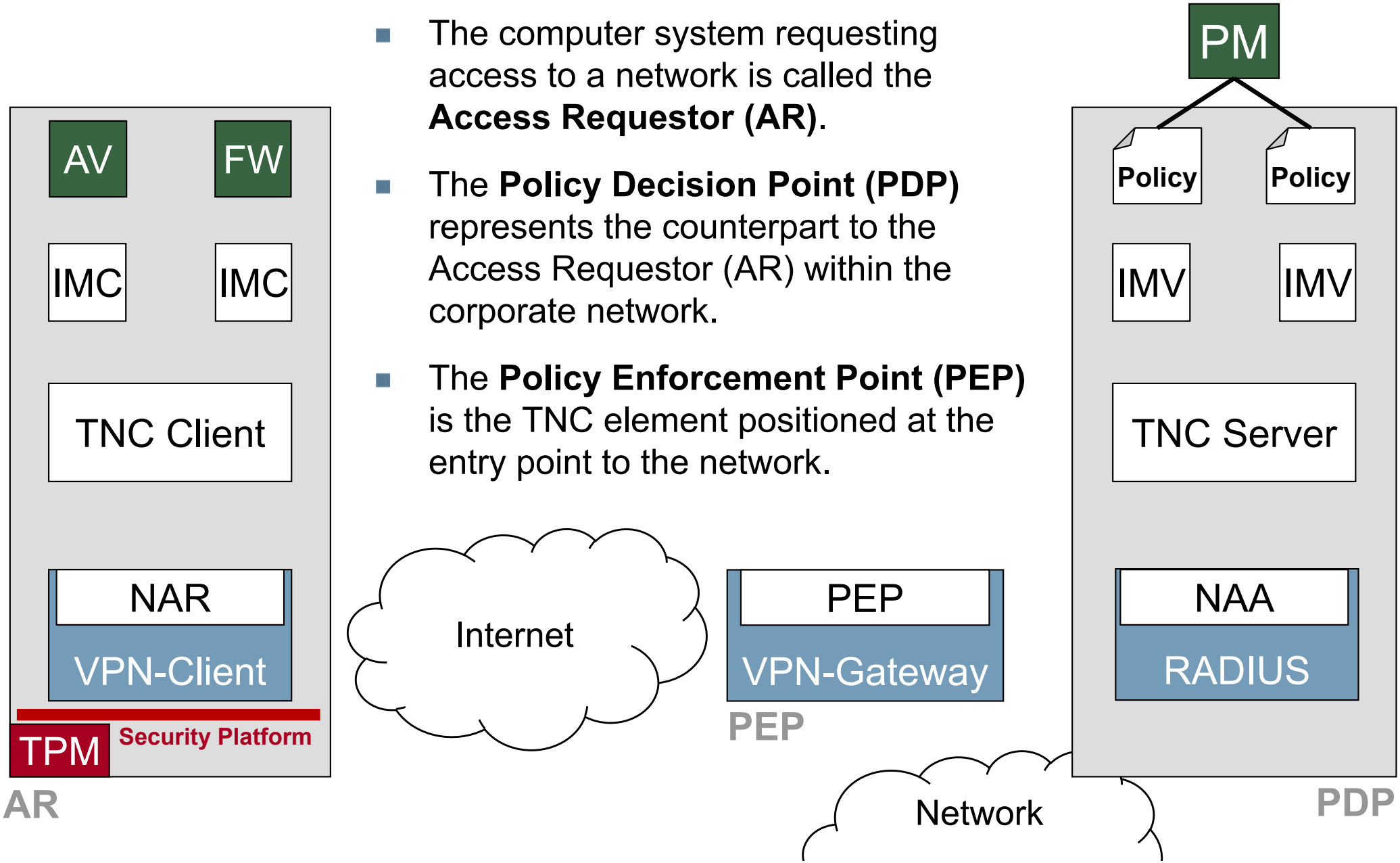
Trusted Network Connect
→ Struktur



© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Overview
→ Trusted Network Connect (TNC)

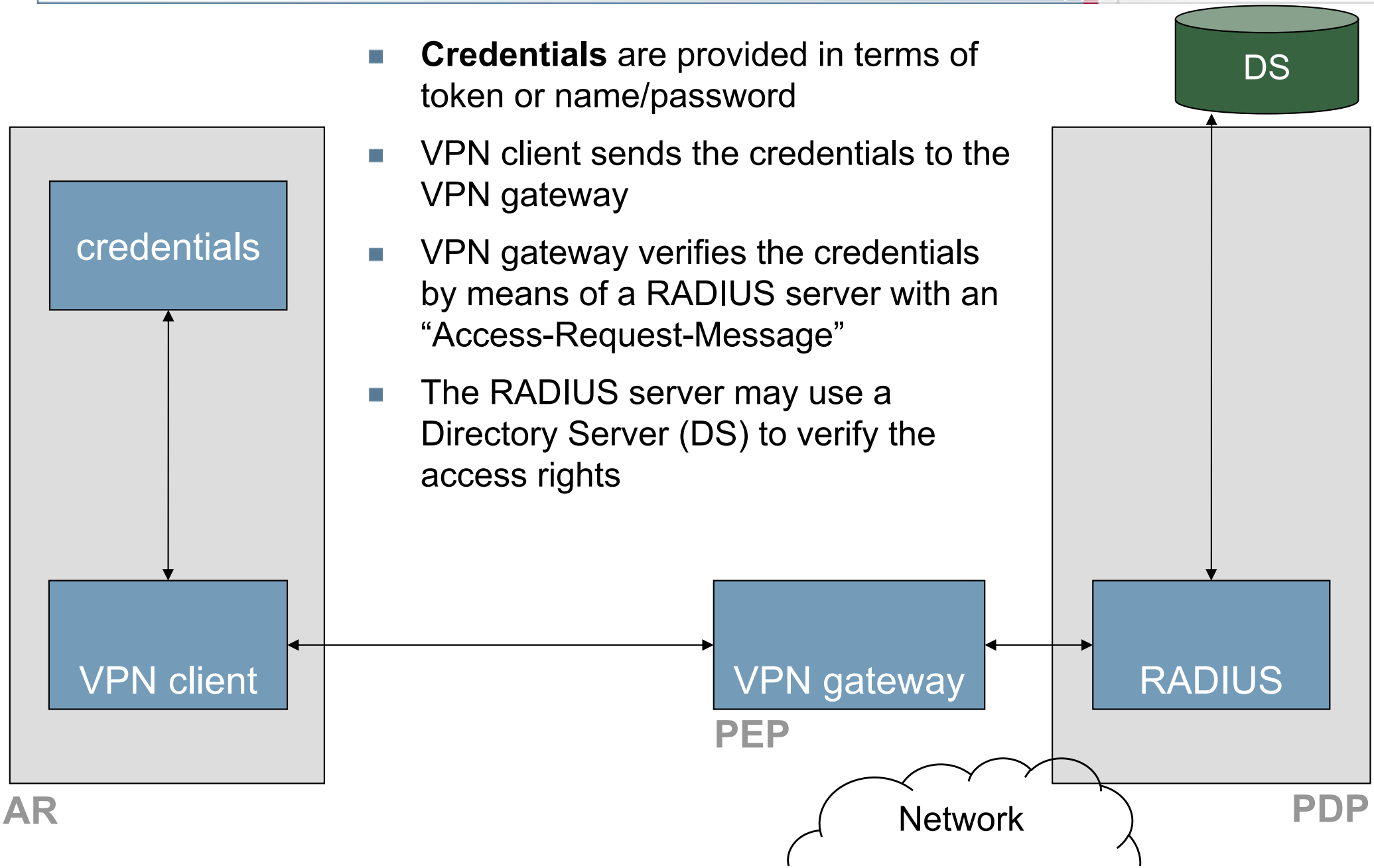
- The computer system requesting access to a network is called the **Access Requestor (AR)**.
- The **Policy Decision Point (PDP)** represents the counterpart to the Access Requestor (AR) within the corporate network.
- The **Policy Enforcement Point (PEP)** is the TNC element positioned at the entry point to the network.



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (1/6)
→ Authentication/authorization (1/3)

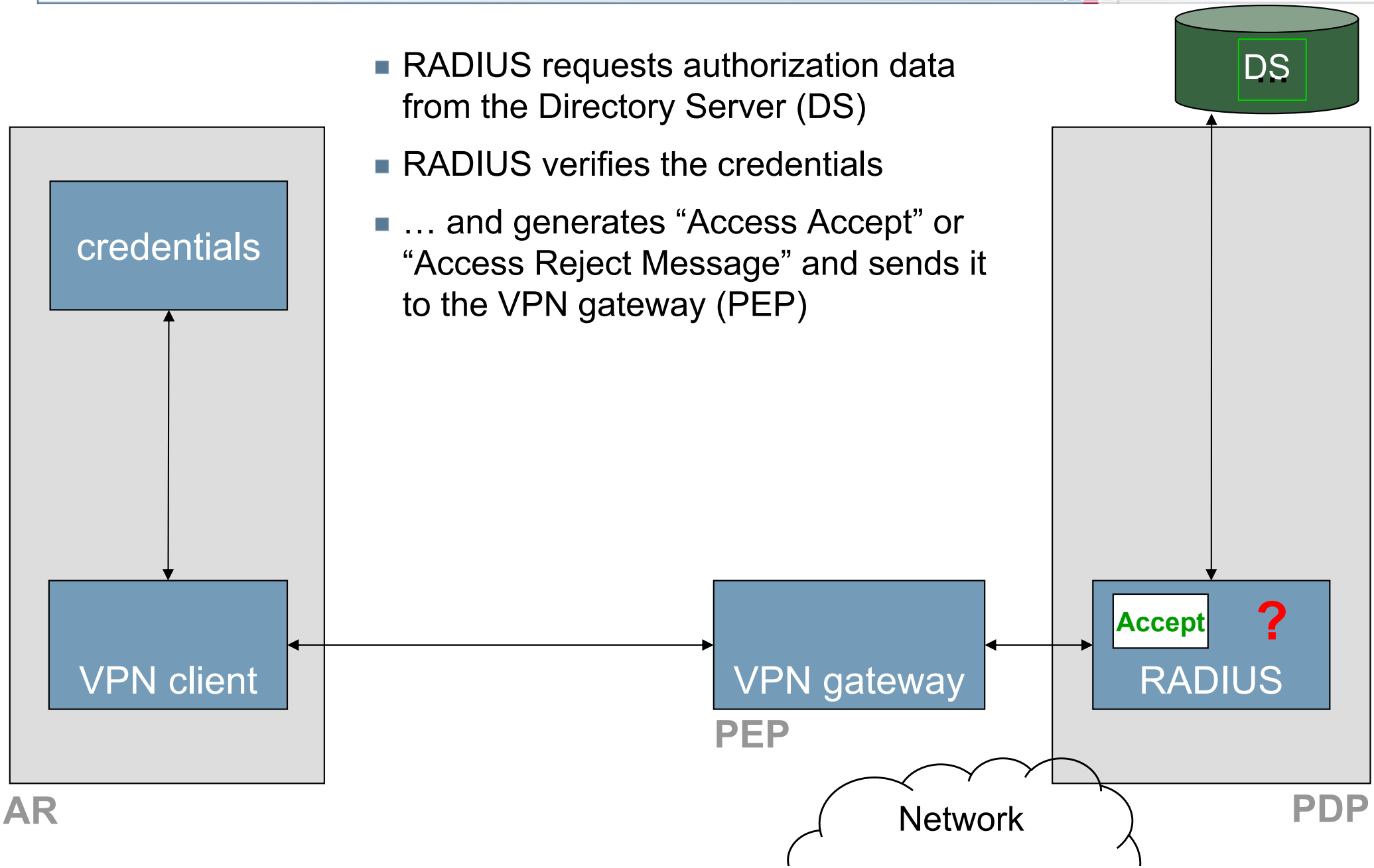
- **Credentials** are provided in terms of token or name/password
- VPN client sends the credentials to the VPN gateway
- VPN gateway verifies the credentials by means of a RADIUS server with an "Access-Request-Message"
- The RADIUS server may use a Directory Server (DS) to verify the access rights



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (2/6)
→ Authentication/authorization (2/3)

- RADIUS requests authorization data from the Directory Server (DS)
- RADIUS verifies the credentials
- … and generates “Access Accept” or “Access Reject Message” and sends it to the VPN gateway (PEP)

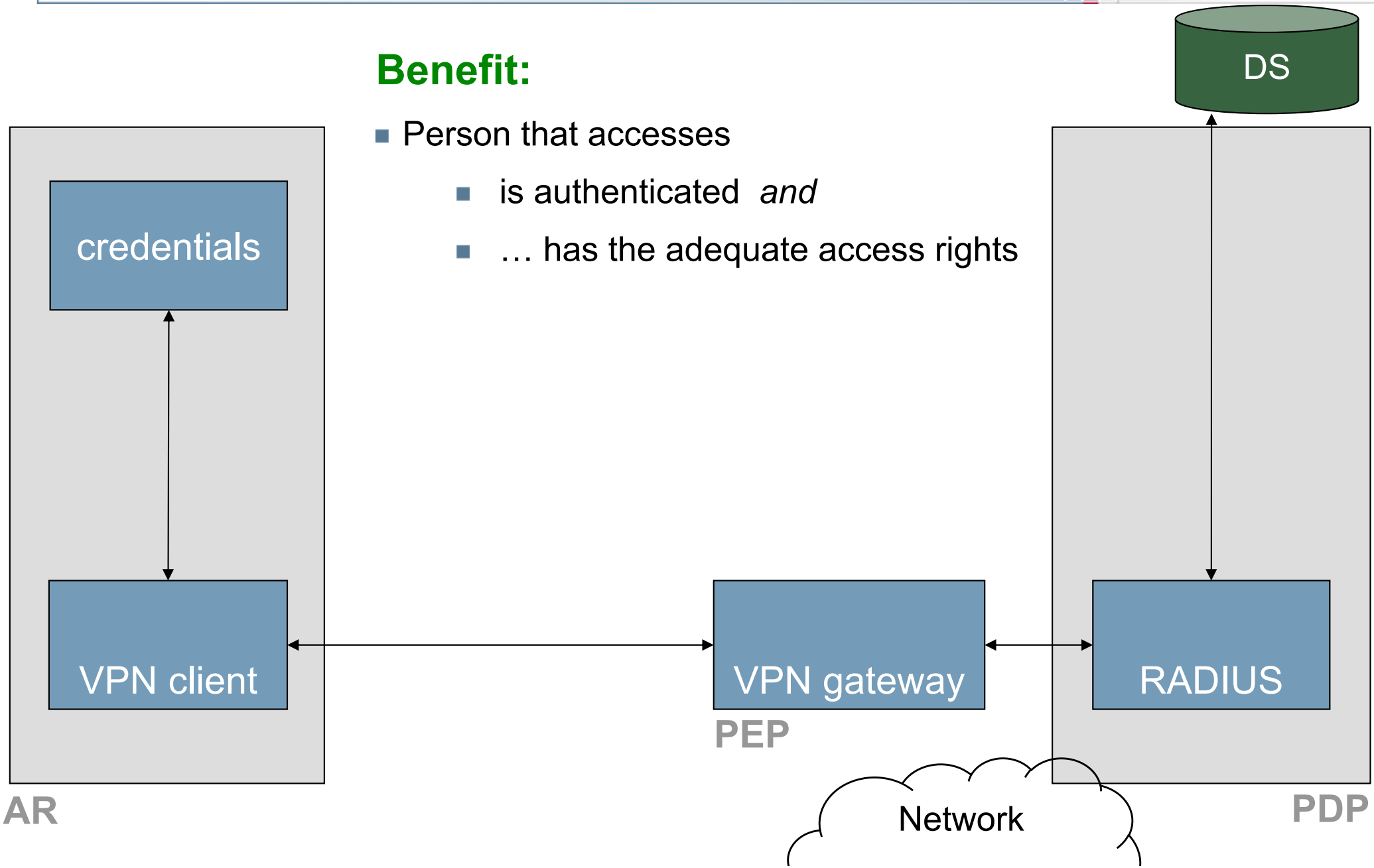


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (3/6)
→ Authentication/authorization (3/3)

Benefit:

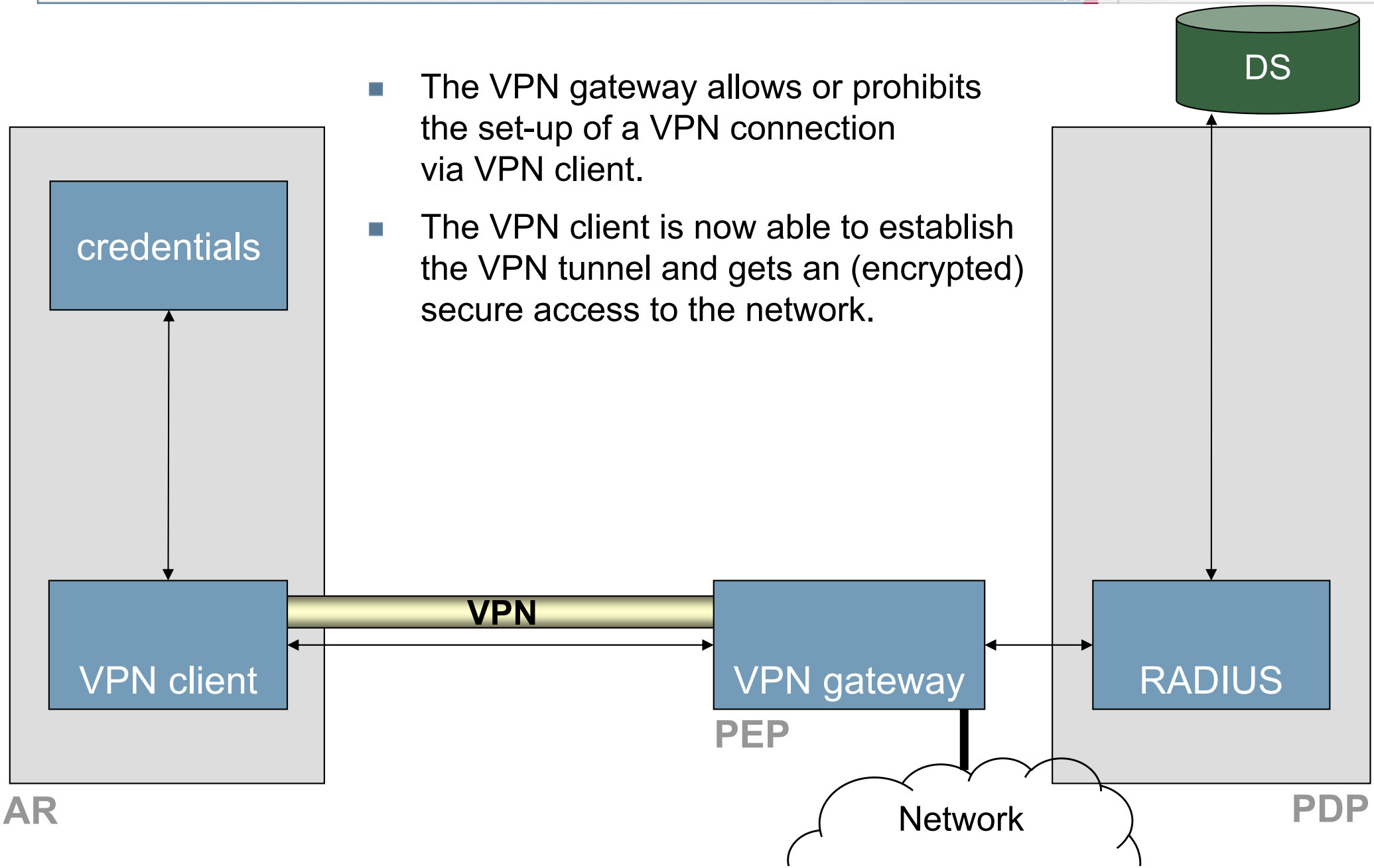
- Person that accesses
  - is authenticated *and*
  - … has the adequate access rights



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (4/6)
→Encrypted communication

- The VPN gateway allows or prohibits the set-up of a VPN connection via VPN client.
- The VPN client is now able to establish the VPN tunnel and gets an (encrypted) secure access to the network.

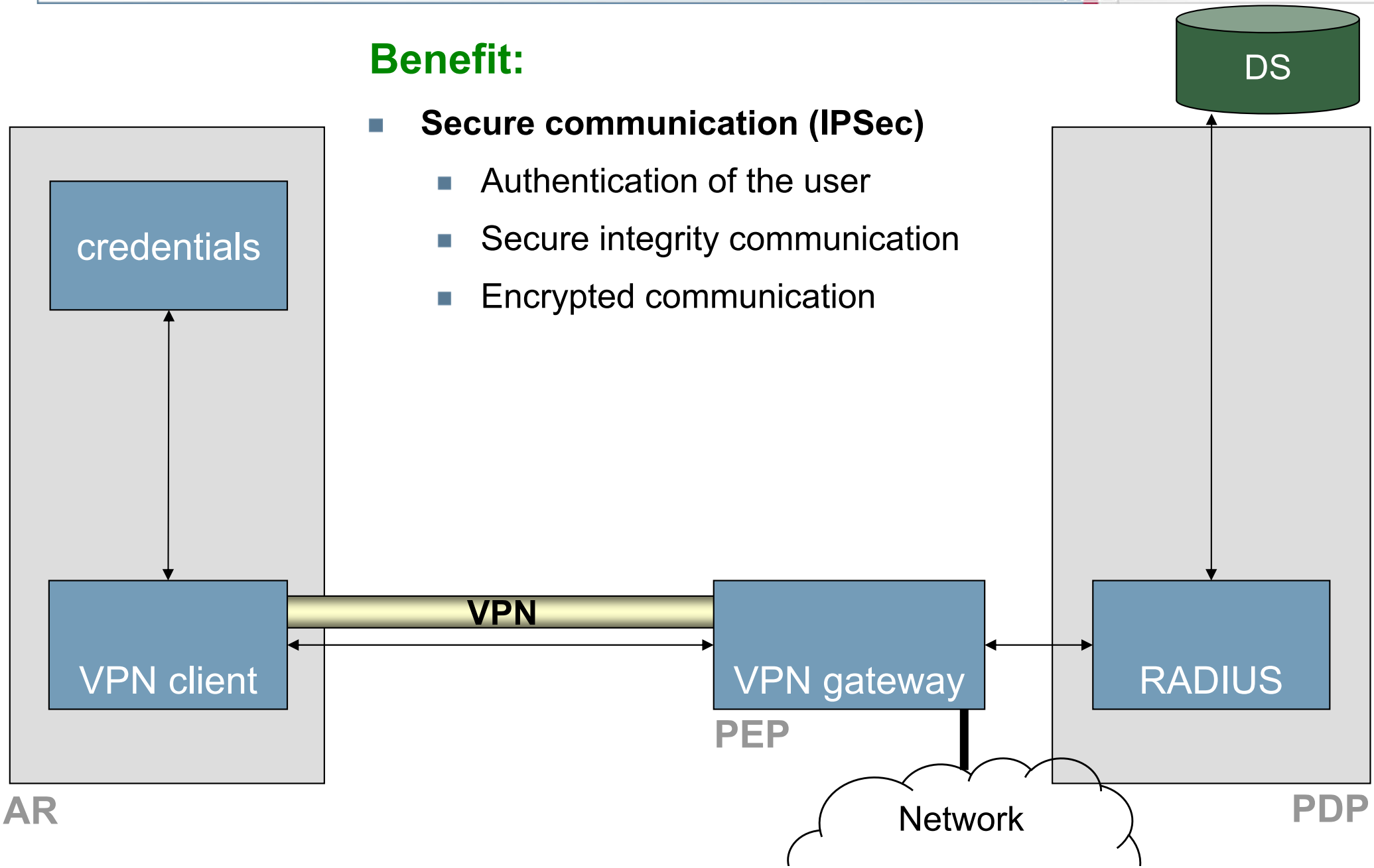


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (5/6)
→ Secure communication (IPSec)

Benefit:

- **Secure communication (IPSec)**
  - Authentication of the user
  - Secure integrity communication
  - Encrypted communication

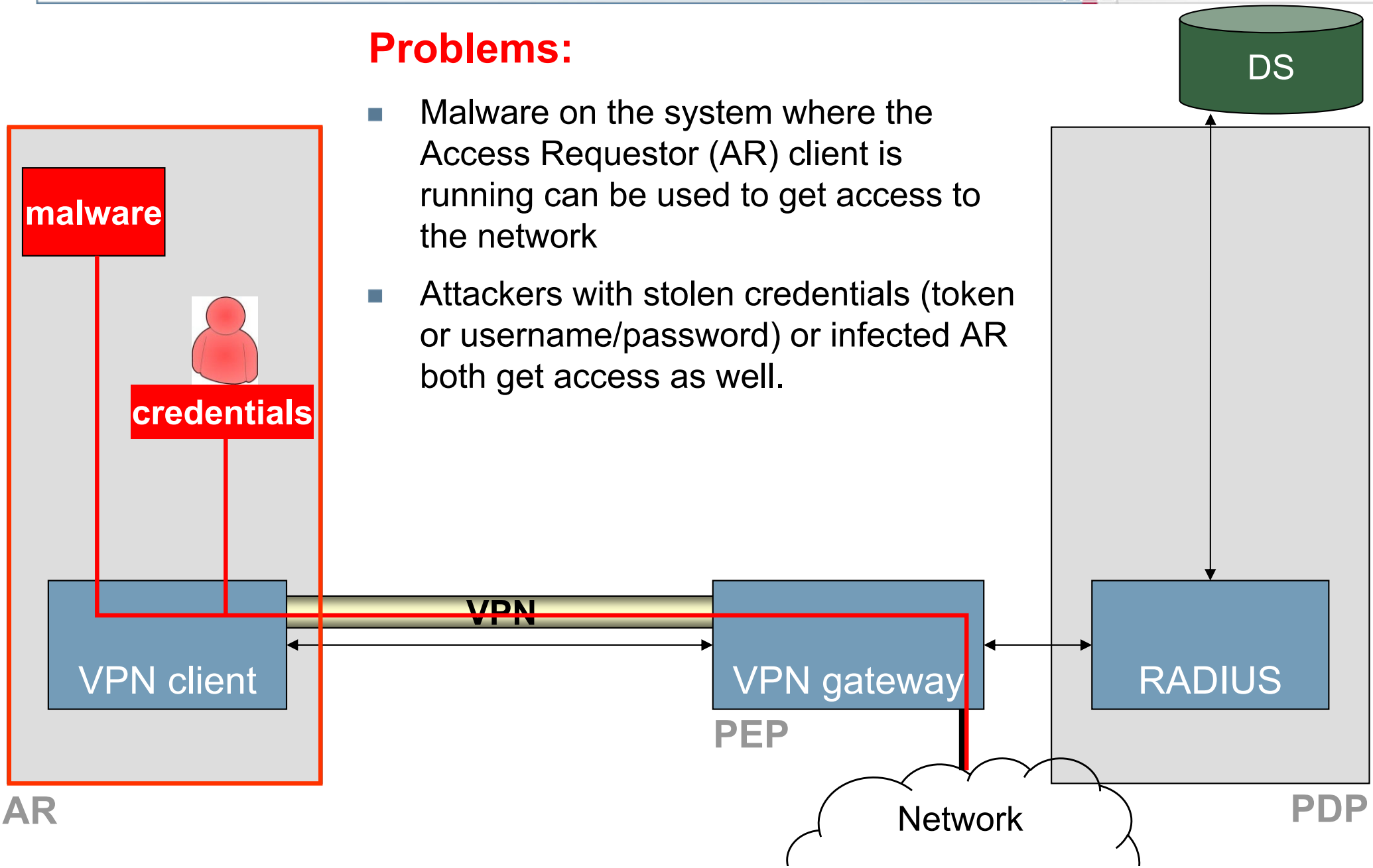


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Communication via VPN (6/6)
→ Open problems with VPN

Problems:

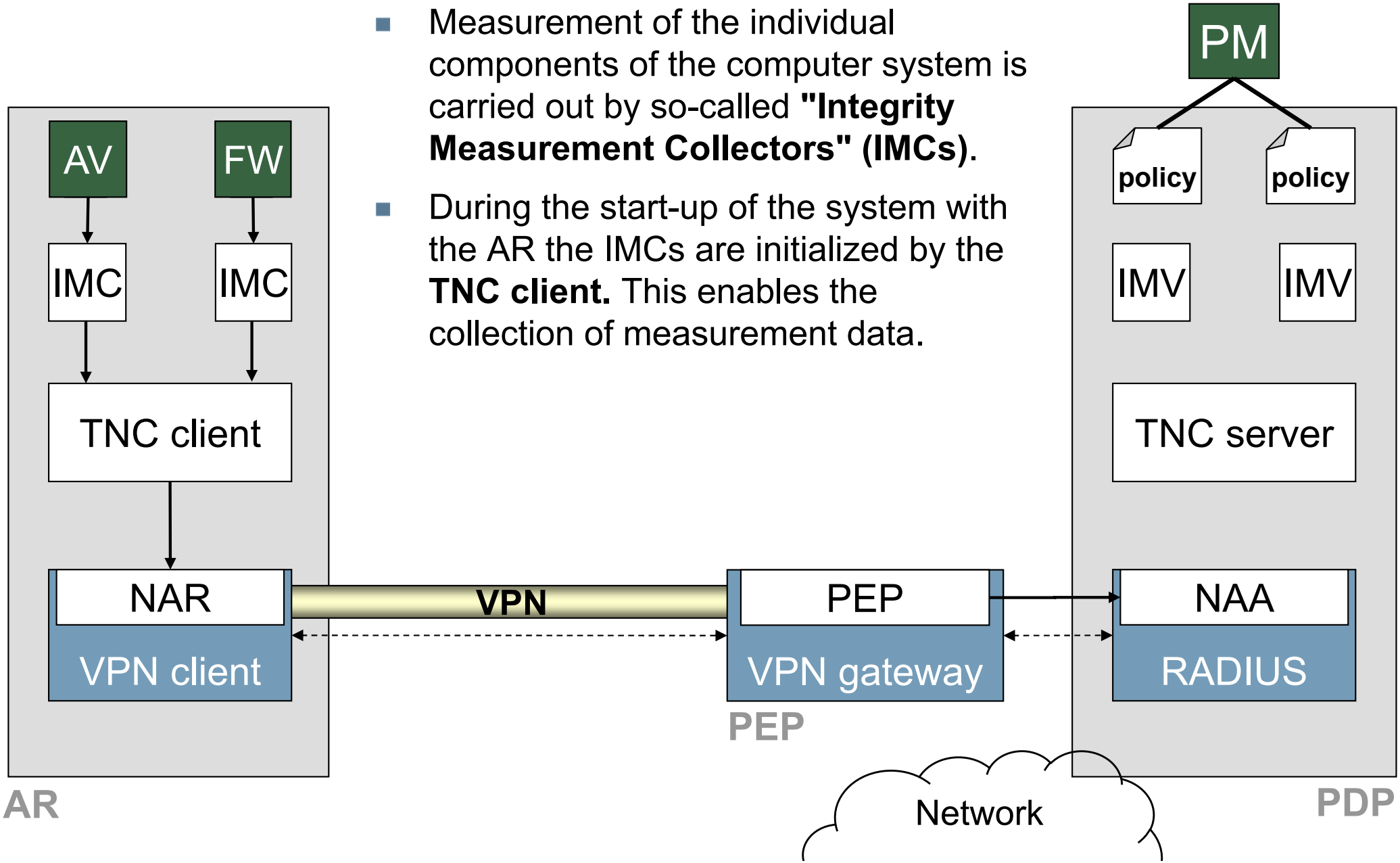
- Malware on the system where the Access Requestor (AR) client is running can be used to get access to the network
- Attackers with stolen credentials (token or username/password) or infected AR both get access as well.



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Trusted Network Connect (TNC)
→ Overview: TNC-functions (1/2)

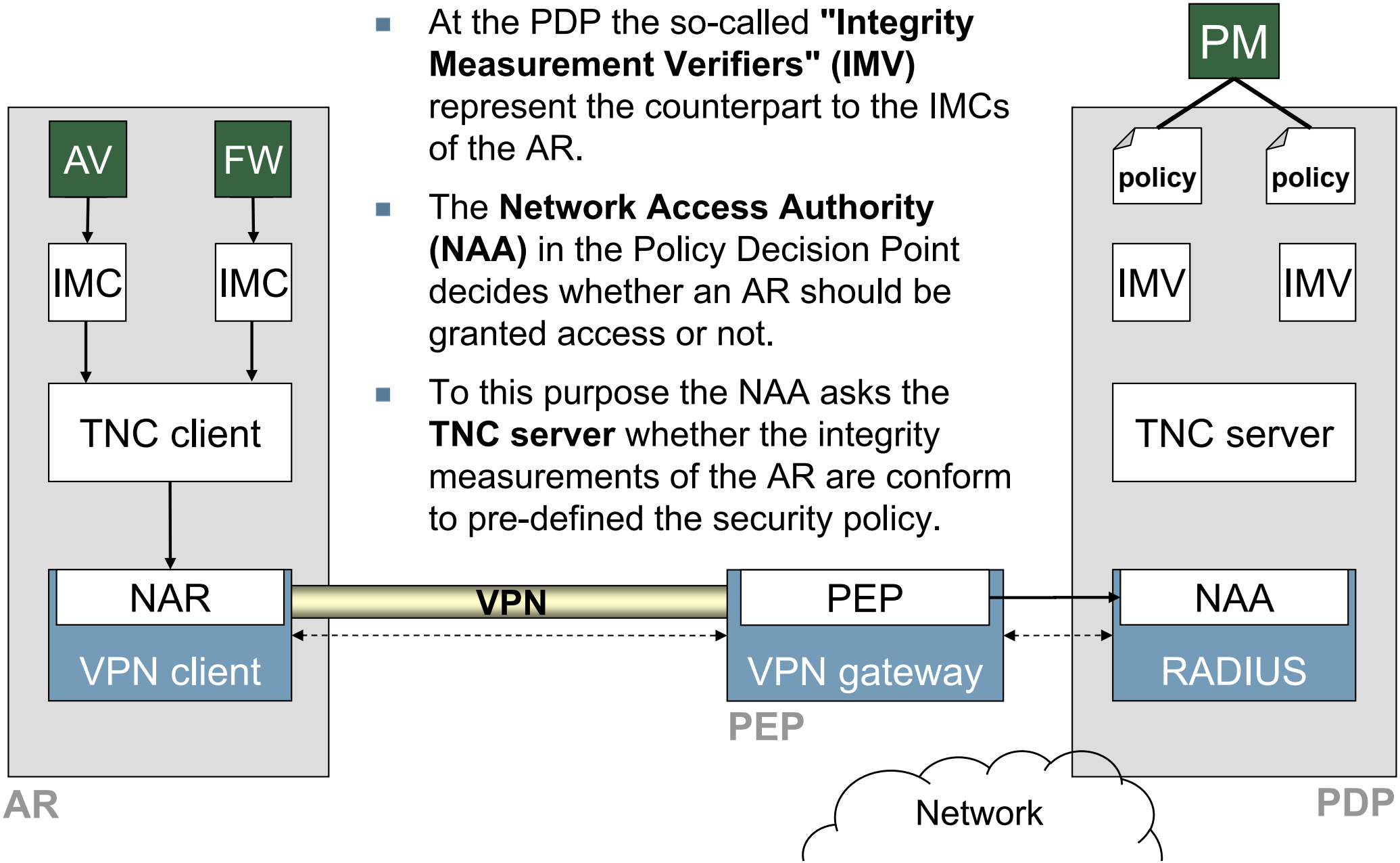
- Measurement of the individual components of the computer system is carried out by so-called **"Integrity Measurement Collectors" (IMCs)**.
- During the start-up of the system with the AR the IMCs are initialized by the **TNC client.** This enables the collection of measurement data.



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Trusted Network Connect (TNC)
→ Overview: TNC-functions (2/2)

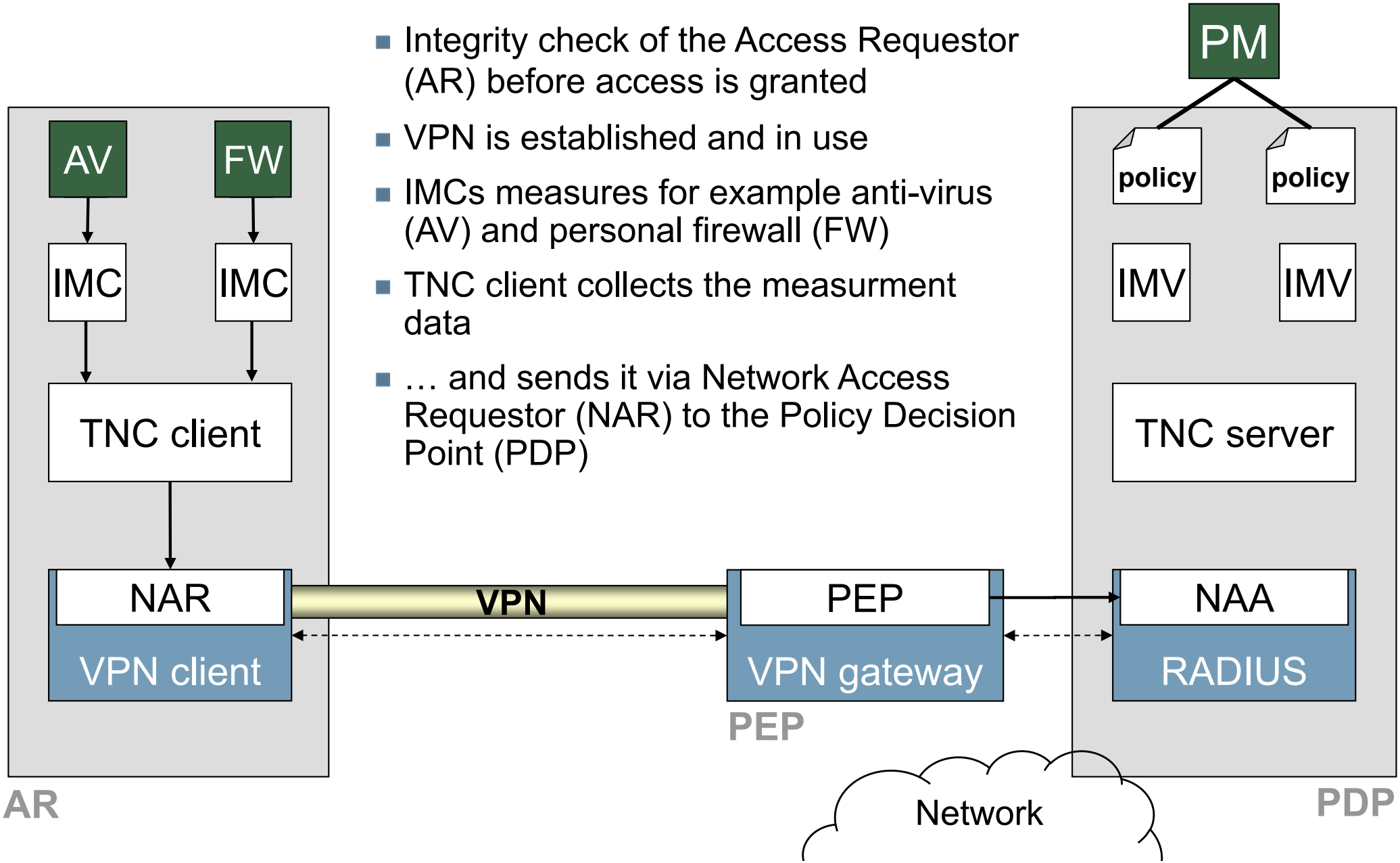
- At the PDP the so-called **"Integrity Measurement Verifiers" (IMV)** represent the counterpart to the IMCs of the AR.
- The **Network Access Authority (NAA)** in the Policy Decision Point decides whether an AR should be granted access or not.
- To this purpose the NAA asks the **TNC server** whether the integrity measurements of the AR are conform to pre-defined the security policy.



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC – phases
→ Assessment phase (1/3)

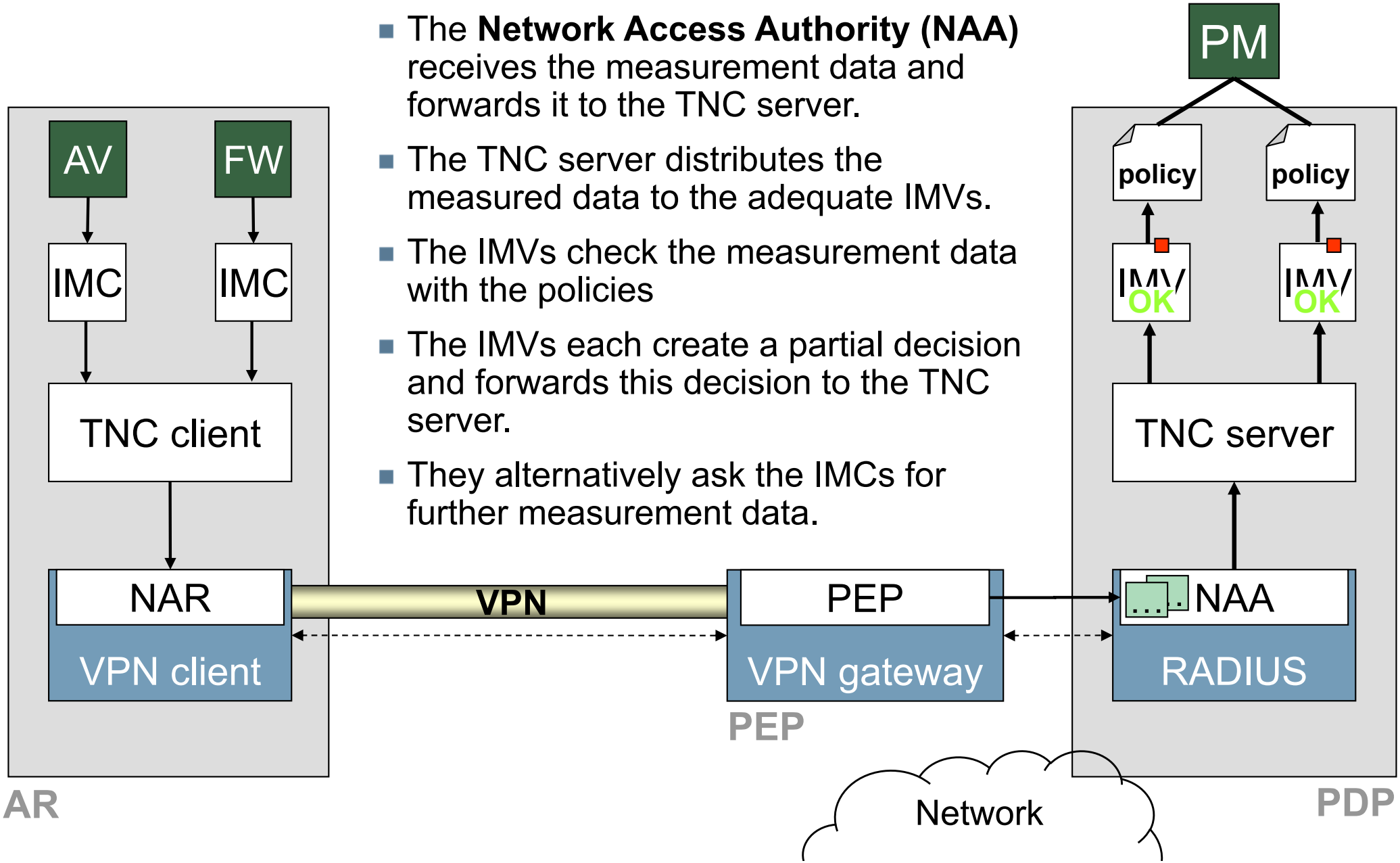
- Integrity check of the Access Requestor (AR) before access is granted
- VPN is established and in use
- IMCs measures for example anti-virus (AV) and personal firewall (FW)
- TNC client collects the measurment data
- … and sends it via Network Access Requestor (NAR) to the Policy Decision Point (PDP)



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC – phases
→ Assessment phase (2/3)

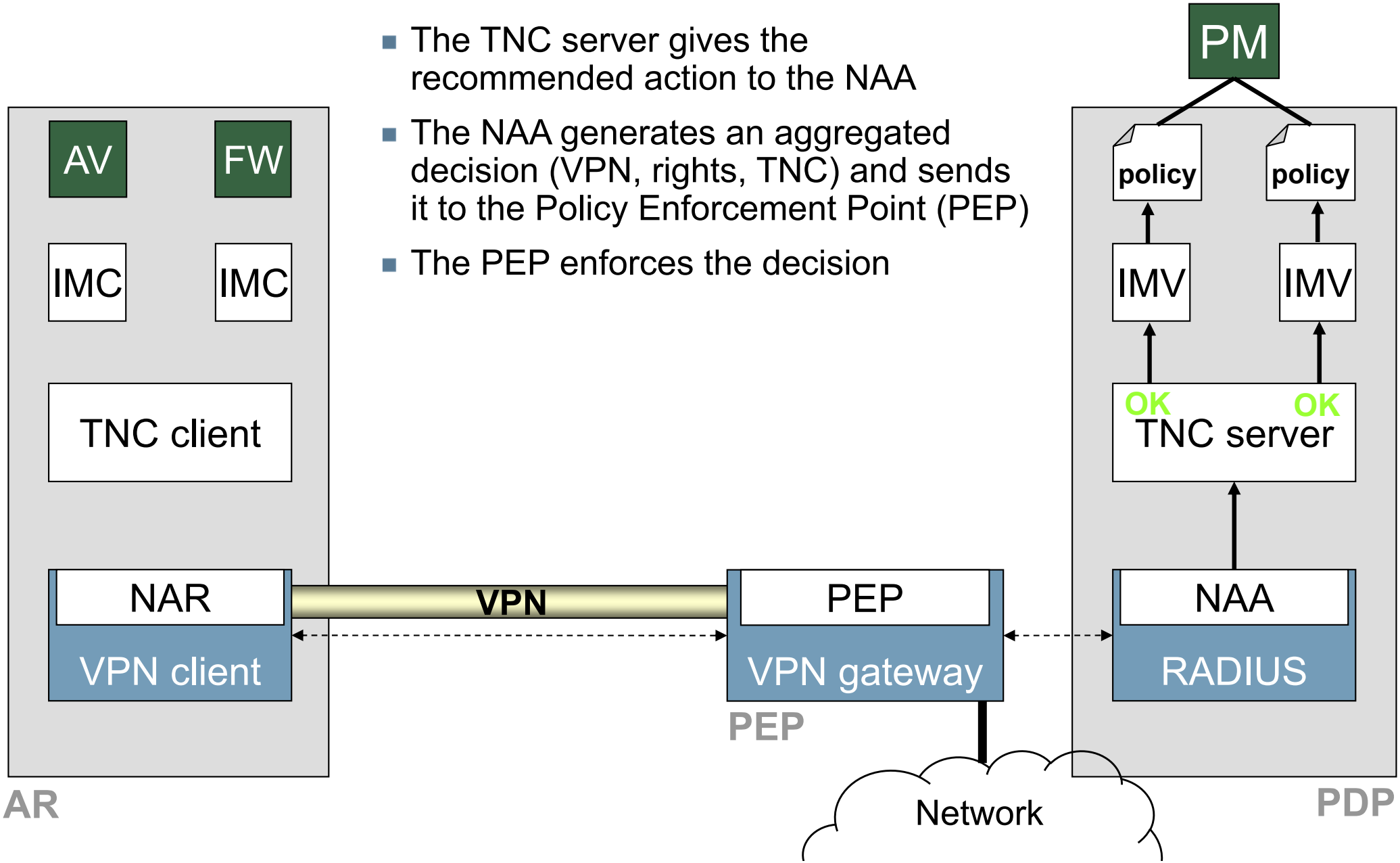
- The **Network Access Authority (NAA)** receives the measurement data and forwards it to the TNC server.
- The TNC server distributes the measured data to the adequate IMVs.
- The IMVs check the measurement data with the policies
- The IMVs each create a partial decision and forwards this decision to the TNC server.
- They alternatively ask the IMCs for further measurement data.



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC – phases
→ Assessment phase (3/3)

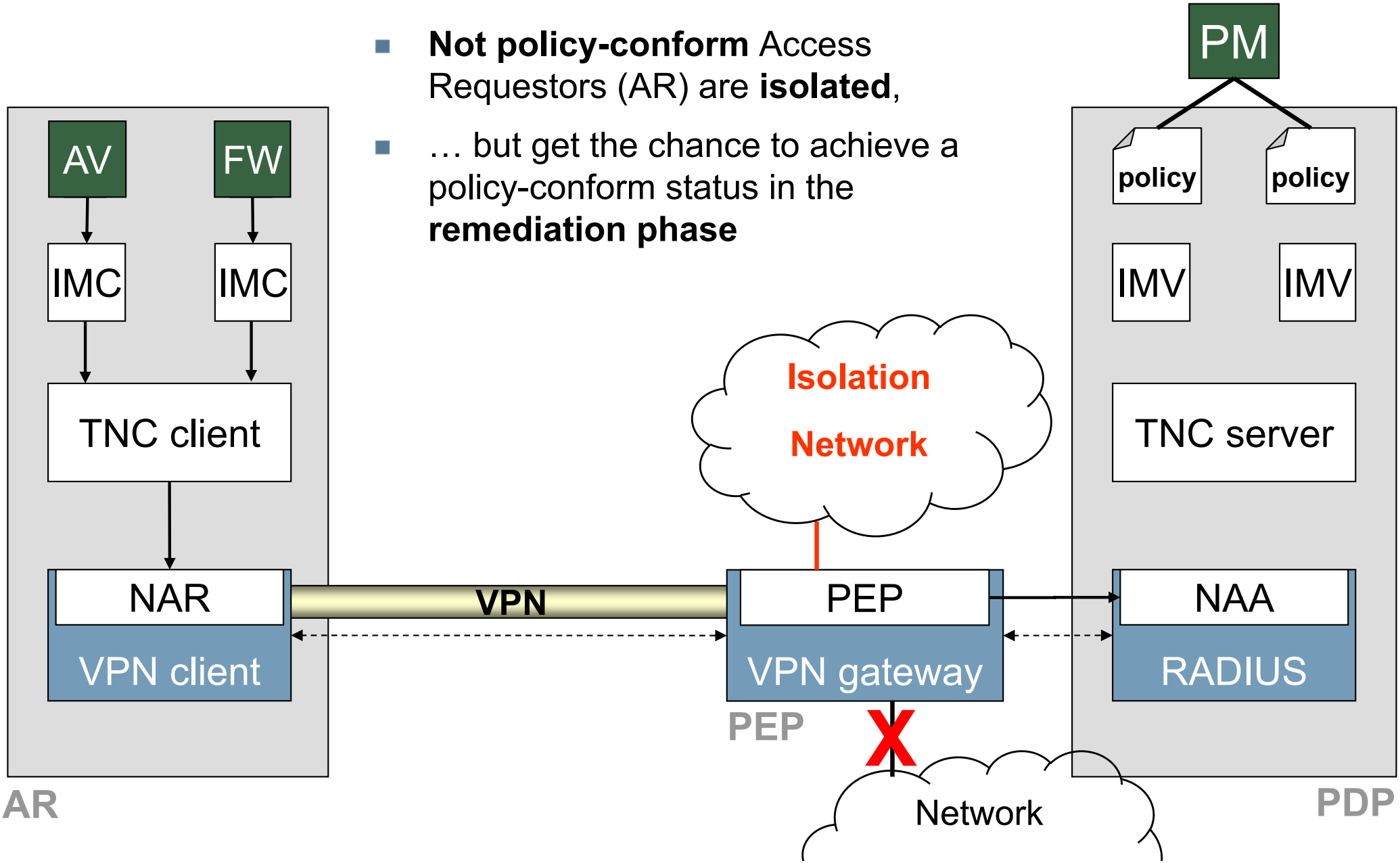
- The TNC server gives the recommended action to the NAA
- The NAA generates an aggregated decision (VPN, rights, TNC) and sends it to the Policy Enforcement Point (PEP)
- The PEP enforces the decision



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC – phases
→ Isolation and remediation phase

- **Not policy-conform** Access Requestors (AR) are **isolated**,
- … but get the chance to achieve a policy-conform status in the **remediation phase**

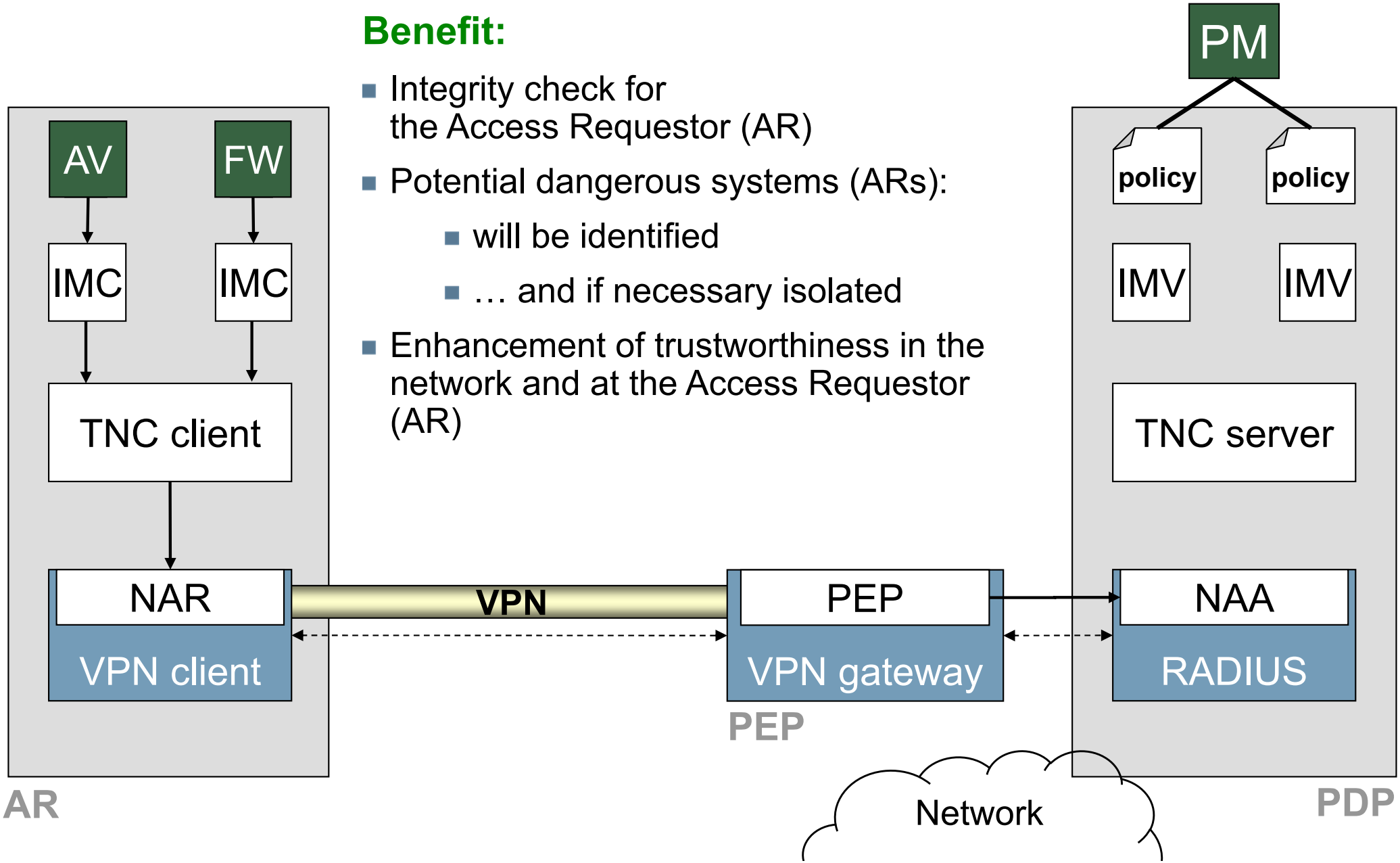


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC
→ Trusted Network Connect

Benefit:

- Integrity check for the Access Requestor (AR)
- Potential dangerous systems (ARs):
  - will be identified
  - … and if necessary isolated
- Enhancement of trustworthiness in the network and at the Access Requestor (AR)

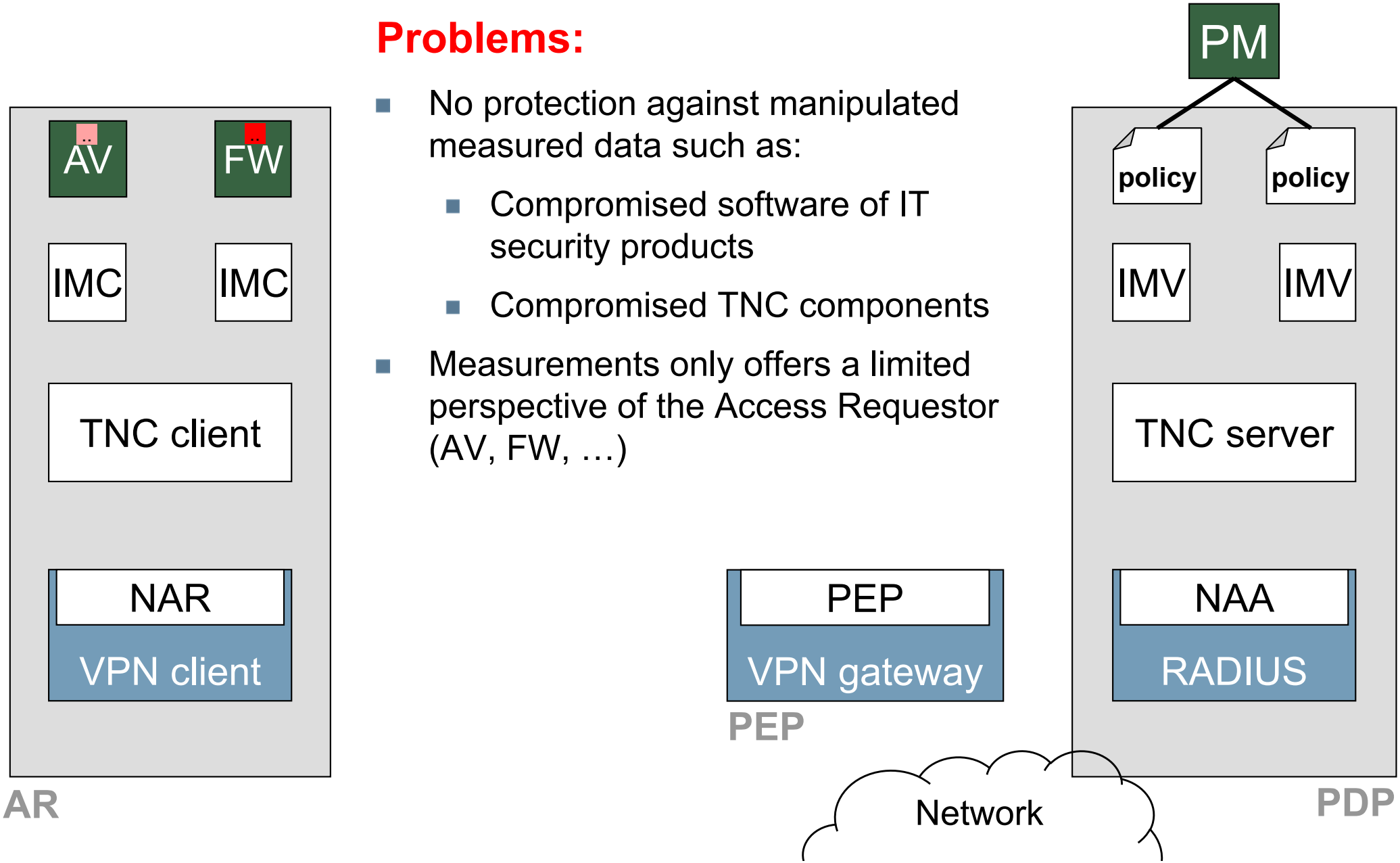


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC
→ Open problems with TNC

Problems:

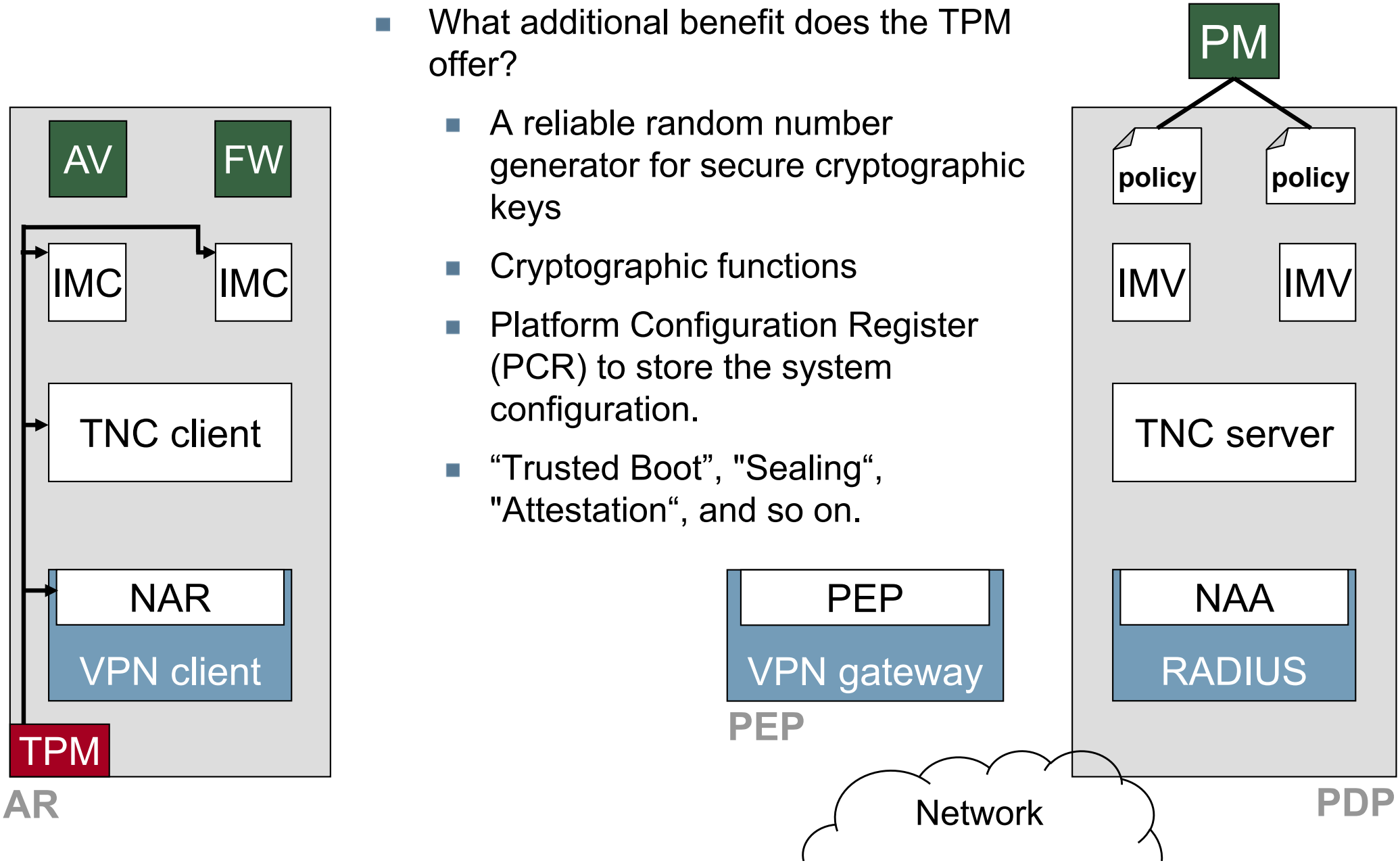
- No protection against manipulated measured data such as:
  - Compromised software of IT security products
  - Compromised TNC components
- Measurements only offers a limited perspective of the Access Requestor (AV, FW, …)



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC+
→ TNC + TPM

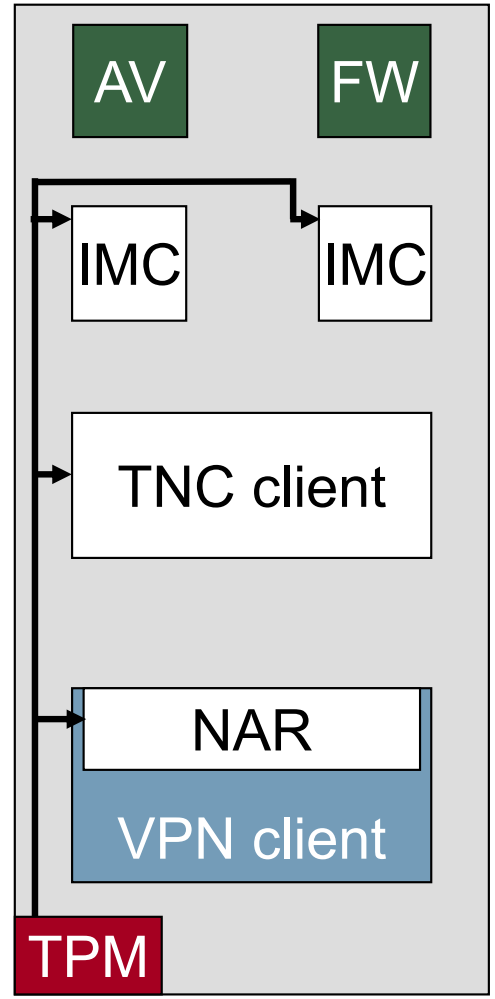
- What additional benefit does the TPM offer?
  - A reliable random number generator for secure cryptographic keys
  - Cryptographic functions
  - Platform Configuration Register (PCR) to store the system configuration.
  - “Trusted Boot”, "Sealing“, "Attestation“, and so on.



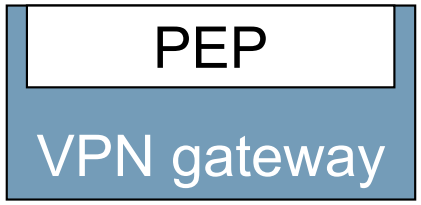
© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC+
→ additional benefit: TPM

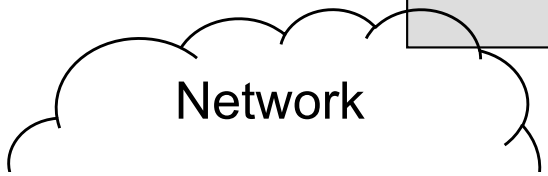
- TPM functions help with:
  - Integrity checks for the TNC components
  - Support the attestation & authentication of the platform
  - Linking communication connections to a platform (against attacks)
  - Protecting cryptographic keys (attestation, VPN, authentication, …)

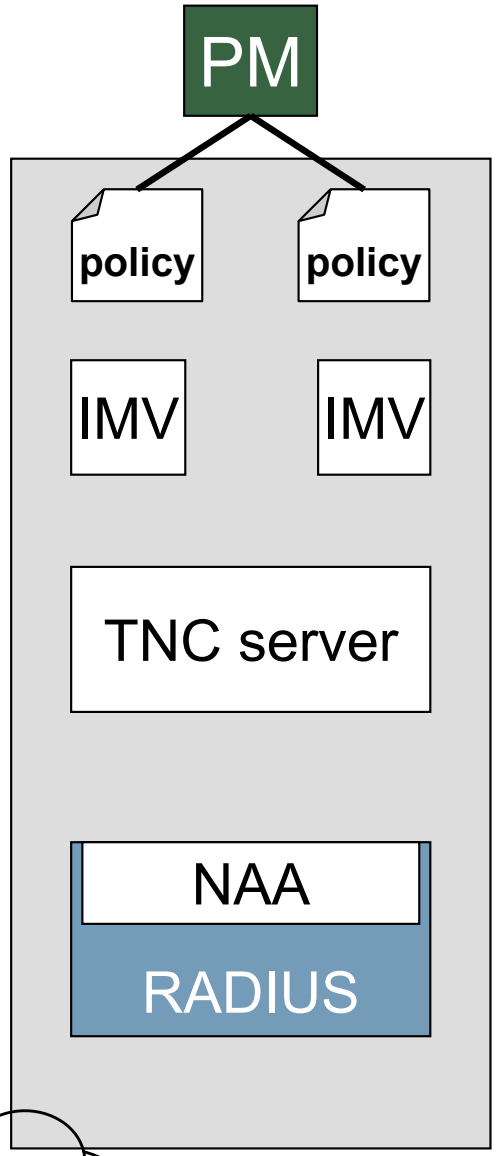


AR



PEP





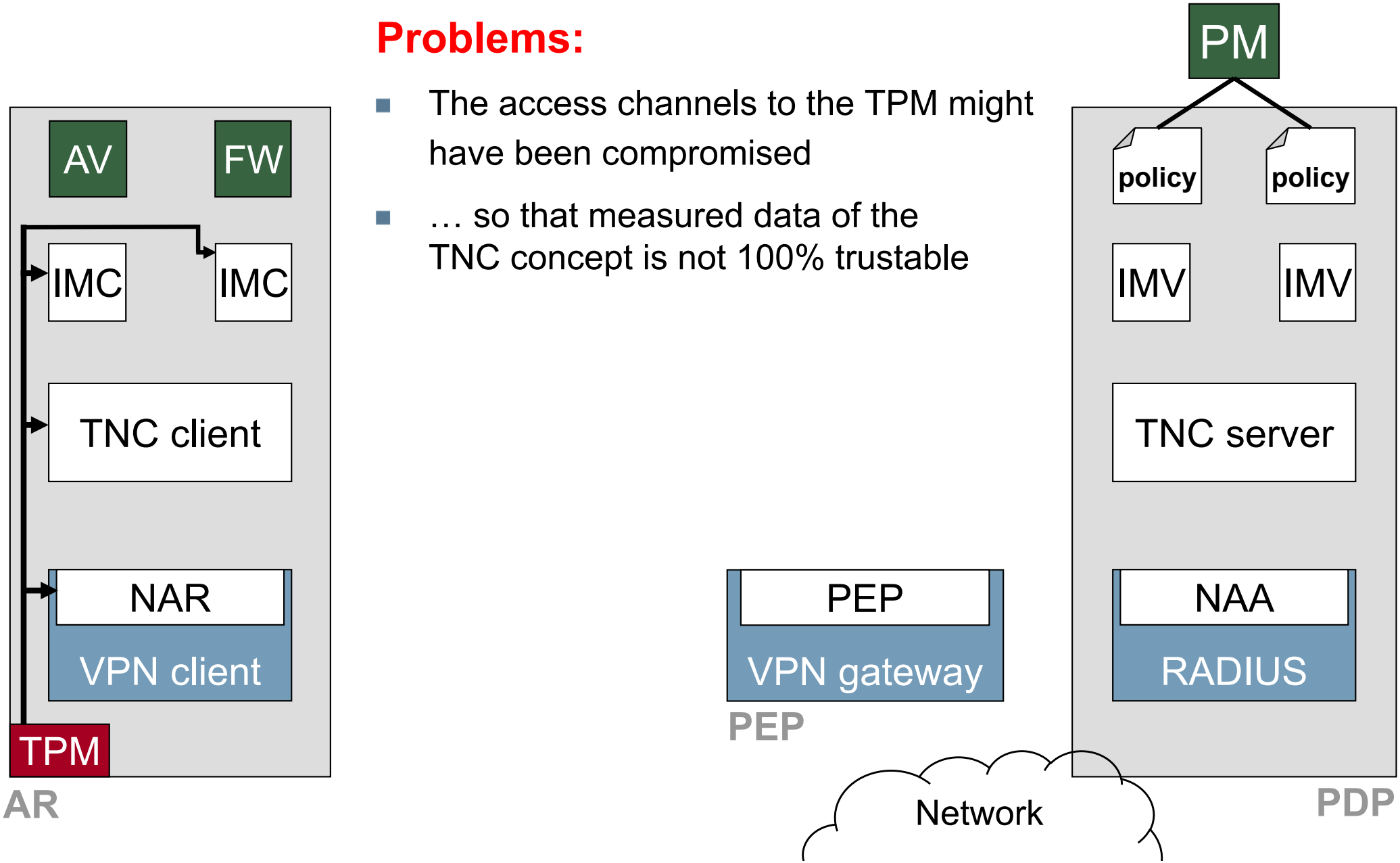
PDP

© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC+
→ TPM: restrictions

Problems:

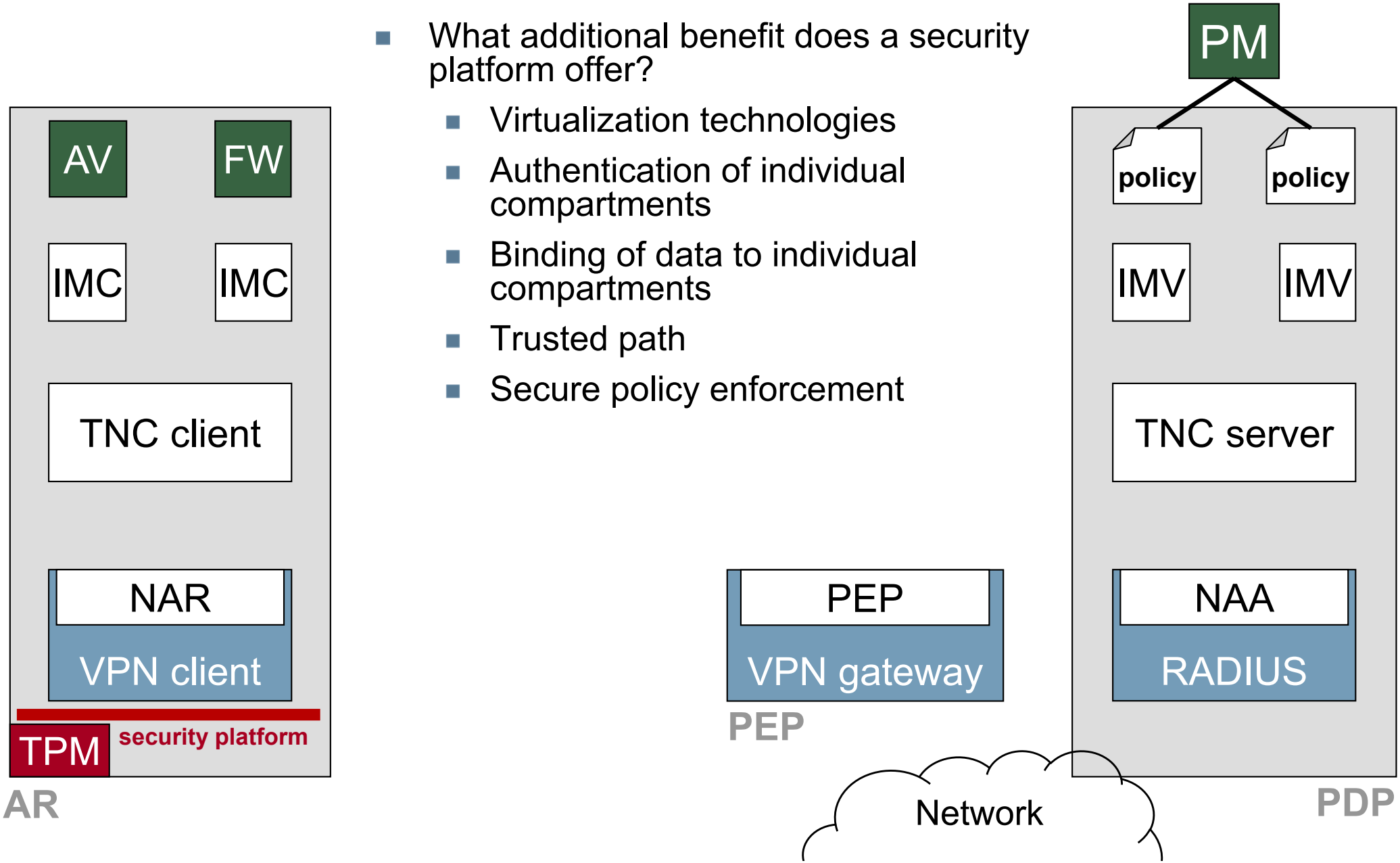
- The access channels to the TPM might have been compromised
- … so that measured data of the TNC concept is not 100% trustable



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC++
→ TNC + TPM + security platform

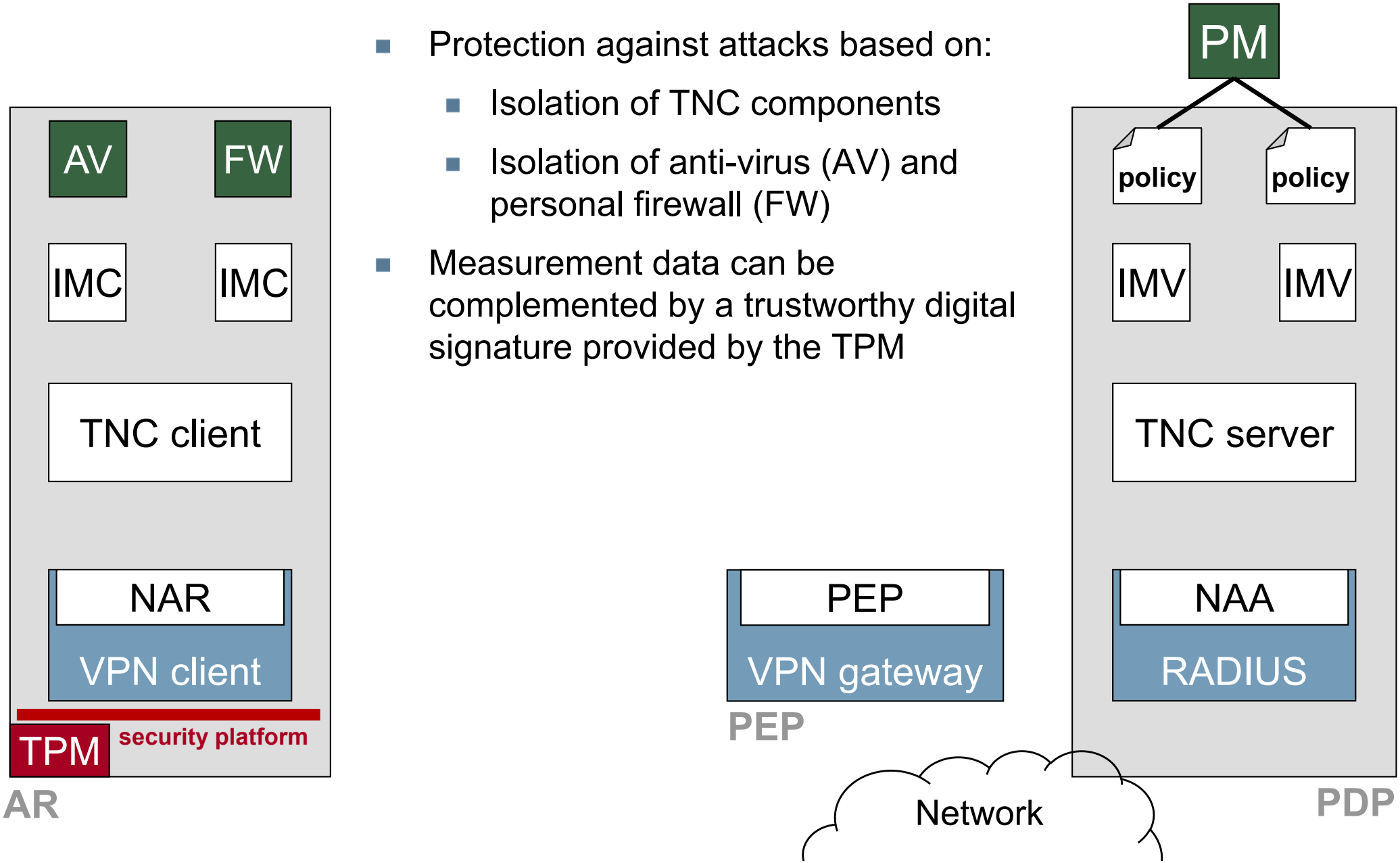
- What additional benefit does a security platform offer?
  - Virtualization technologies
  - Authentication of individual compartments
  - Binding of data to individual compartments
  - Trusted path
  - Secure policy enforcement



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC++
→ Overvalue: security platform

- Protection against attacks based on:
  - Isolation of TNC components
  - Isolation of anti-virus (AV) and personal firewall (FW)
- Measurement data can be complemented by a trustworthy digital signature provided by the TPM

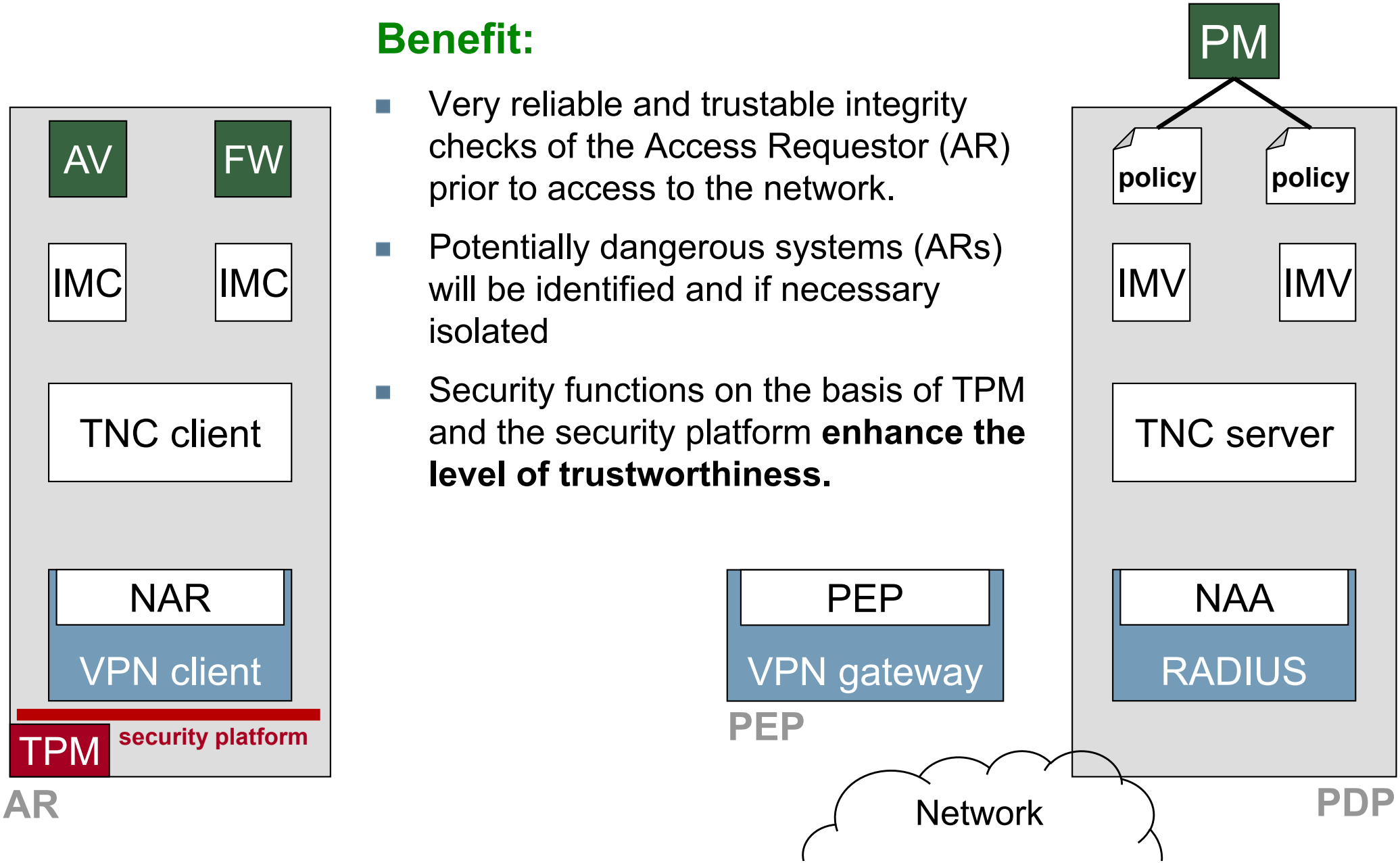


© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

TNC++
→ Added value: security platform

Benefit:

- Very reliable and trustable integrity checks of the Access Requestor (AR) prior to access to the network.
- Potentially dangerous systems (ARs) will be identified and if necessary isolated
- Security functions on the basis of TPM and the security platform **enhance the level of trustworthiness.**



© Prof. Dr. Norbert Pohlmann, Institute for Internet Security - if(is), University of Applied Sciences Gelsenkirchen, Germany

Festlegung einer sicheren
→ Systemkonfiguration (1/2)

- Mit Trusted Computing können Hard- und Software-Systemkonfigurationen gemessen und jederzeit überprüft werden.
- Eine wichtige Frage ist aber, wer definiert, was eine sichere und vertrauenswürdige Systemkonfiguration ist.

- **Der Hersteller der IT-Systems?**
  - Der Hersteller **kennt die Hardware sehr gut**, weil er sie selber baut oder bauen lässt.
  - Er ist für das **Betriebssystemen** und **Basis-Anwendungen** verantwortlich.
  - Mit diesem Wissen und Gestaltungsspielraum kann jeder **Hersteller** genau **festlegen**, **was** eine **sichere** und **vertrauenswürdige** Systemkonfiguration **ist**.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Festlegung einer sicheren
→ Systemkonfiguration (2/2)

- **Der Hersteller des Cyber-Sicherheitssystems?**
  - Er kennt sich mit den **Cyber-Sicherheitsproblemen** und **Angriffsvektoren** aus und kann auf dieser Basis definieren, was eine sichere und vertrauenswürdige Systemkonfiguration ist.

- **Das Anwendungsunternehmen der IT-Systeme?**
  - Das Anwendungsunternehmen hat **Erfahrungen mit der Nutzung** der IT-Systeme, kennt die **realen Angriffe** und die daraus **verursachten Schäden**.
  - Mit diesen praktischen Erfahrungen kann das Anwendungsunternehmen sichere und vertrauenswürdige Systemkonfigurationen definieren.

- **Kooperation der unterschiedlichen Rollen**
  - Ideal wäre es, wenn sich die **Hersteller** der IT-Systeme und Cyber-Sicherheitssysteme sowie die **Anwendungsunternehmen** zusammentun würden, um eine sichere und vertrauenswürdige Systemkonfigurationen der genutzten IT-Systeme zu definieren.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einleitung
- IT-Sicherheitsarchitektur
- Trusted Computing-Funktion
- Trusted Platform - Turaya
- Trusted Network Connect - TNC
- **Zusammenfassung**

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Zusammenfassung (1/2)

- Die Kernfunktionalitäten von Trusted Computing sind:
  - **Robustheit und Modularität**
  - **Integritätsüberprüfung**
  - **Trusted Process**
  - **Trusted Plattform**

- Vor- und Nachteile der verschiedenen **Kernelarchitekturen** müssen miteinander abgewogen werden.

- **CRTM** ist die Vertrauensbasis. Das Vertrauen ist **transitiv**.

- Die **TPM Schlüsselhierarchie** ermöglicht eine sichere Speicherung von Daten, auch auf externen Speichermedien.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Trusted Computing
→ Zusammenfassung (2/2)

- Wichtige Trusted Computing Funktionen sind:
  - **Authenticated Boot**
  - **Binding**
  - **Sealing**
  - (remote) **Attestation**
- Vertrauenswürdige Netzwerkverbindungen können durch **Trusted Network Connect (TNC)** realisiert werden.
- Die Festlegung einer sicheren und vertrauenswürdigen **Systemkonfiguration** ist mit zahlreichen Schwierigkeiten (technisch und politisch) verbunden.

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Trusted Computing

Prof. Dr. (TU NN)
Norbert Pohlmann

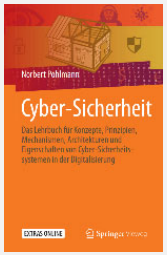
Institut für Internet-Sicherheit – if(is)
Westfälische Hochschule, Gelsenkirchen
http://www.internet-sicherheit.de

if(is) internet-sicherheit.

Anhang / Credits

Wir empfehlen

- **Cyber-Sicherheit**
  Das **Lehrbuch** für Konzepte, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung“, Springer Vieweg Verlag, Wiesbaden 2022
  https://norbert-pohlmann.com/cyber-sicherheit/



- **7. Sinn im Internet (Cyberschutzraum)**
  https://www.youtube.com/cyberschutzraum

CYBER SCHUTZRAUM

- **Master Internet-Sicherheit**
  https://it-sicherheit.de/master-studieren/

MASTER internet-sicherheit.

Besuchen und abonnieren Sie uns :-)

WWW
https://www.internet-sicherheit.de

Facebook
https://www.facebook.com/Internet.Sicherheit.ifis

Twitter
https://twitter.com/_ifis

https://twitter.com/ProfPohlmann

YouTube
https://www.youtube.com/user/InternetSicherheitDE/

Prof. Norbert Pohlmann
https://norbert-pohlmann.com/

Quellen Bildmaterial

Eingebettete Piktogramme:
- Institut für Internet-Sicherheit – if(is)

Der Marktplatz IT-Sicherheit

(IT-Sicherheits-) Anbieter, Lösungen, Jobs, Veranstaltungen und Hilfestellungen (Ratgeber, IT-Sicherheitstipps, Glossar, u.v.m.) leicht & einfach finden.
https://www.it-sicherheit.de/

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen

Literatur
→ Artikel / Bücher

N. Pohlmann, A.-R. Sadeghi, C. Stüble: „European Multilateral Secure Computing Base", DuD Datenschutz und Datensicherheit – Recht und Sicherheit in Informationsverarbeitung und Kommunikation, Vieweg Verlag, 09/2004
https://norbert-pohlmann.com/wp-content/uploads/2015/08/148-European-Multilateral-Secure-Computing-Base.pdf

M. Jungbauer, N. Pohlmann: „Vertrauenswürdige Netzwerkverbindungen mit Trusted Computing - Sicher vernetzt?“, IT-Sicherheit – Management und Praxis, DATAKONTEXT-Fachverlag, 06/2006
https://norbert-pohlmann.com/wp-content/uploads/2015/08/188-Vertrauenswürdige-Netzwerkverbindungen-mit-Trusted-Computing-–-Sicher-vernetzt-Prof.-Norbert-Pohlmann.pdf

M. Linnemann, N. Pohlmann: „Turaya - Die offene Trusted Computing Sicherheitsplattform", in "Open Source Jahrbuch 2007", Hrsg.: B. Lutterbeck, …, Lehmanns Media, Berlin, 2007
https://norbert-pohlmann.com/wp-content/uploads/2015/08/181-Turaya-–-An-Open-Trusted-Computing-Platform-Prof.-Pohlmann.pdf

M. Jungbauer, N. Pohlmann: „Integrity Check of Remote Computer Systems - Trusted Network Connect". In Proceedings of the ISSE/SECURE 2007 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe/Secure 2007 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Vieweg Verlag, Wiesbaden 2007
https://norbert-pohlmann.com/wp-content/uploads/2015/08/190-Integrity-Check-of-Remote-Computer-Systems-–-Trusted-Network-Connect-Prof.-Norbert-Pohlmann.pdf

N. Pohlmann, A. Speier: „Eine Diskussion über Trusted Computing – Sicherheitsgewinn durch vertrauenswürdige IT-Systeme“, IT-Sicherheit – Management und Praxis, DATAKONTEXT-Fachverlag, 5/2013
https://norbert-pohlmann.com/wp-content/uploads/2015/08/306-Eine-Diskussion-über-Trusted-Computing-Sicherheitsgewinn-durch-vertrauenswürdige-IT-Systeme-Prof-Norbert-Pohlmann.pdf

N. Pohlmann: "Cyber-Sicherheit – Das Lehrbuch für Konzepte, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung“, ISBN 978-3-658-25397-4; 594 Seiten, Springer-Vieweg Verlag, Wiesbaden 2019
https://norbert-pohlmann.com/cyber-sicherheit/

© Prof. Norbert Pohlmann, Institut für Internet-Sicherheit - if(is), Westfälische Hochschule, Gelsenkirchen