

Ransomware

Hochschule für Technik Stuttgart – Aktuelle Themen der IT-Sicherheit

23.01.2026

Agenda

Aktuelle Bedrohungslage

Ziele & Motivation

Arten von Ransomware

Case Study: Edu-Ransomware

Technische Architektur

Die Phasen eines Angriffs

Gegenmaßnahmen

Fazit

Aktuelle Bedrohungslage

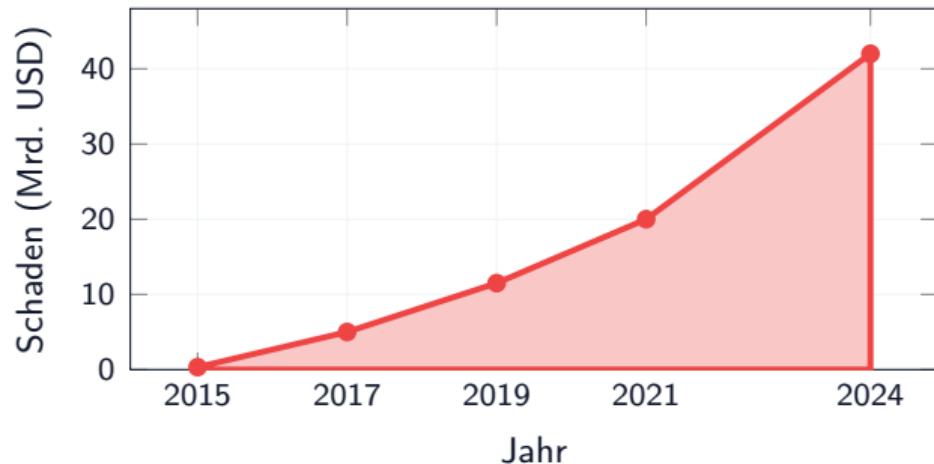
Finanzielle Auswirkungen

Geschätzte Weltweite Schäden

2017: 5 Mrd. USD

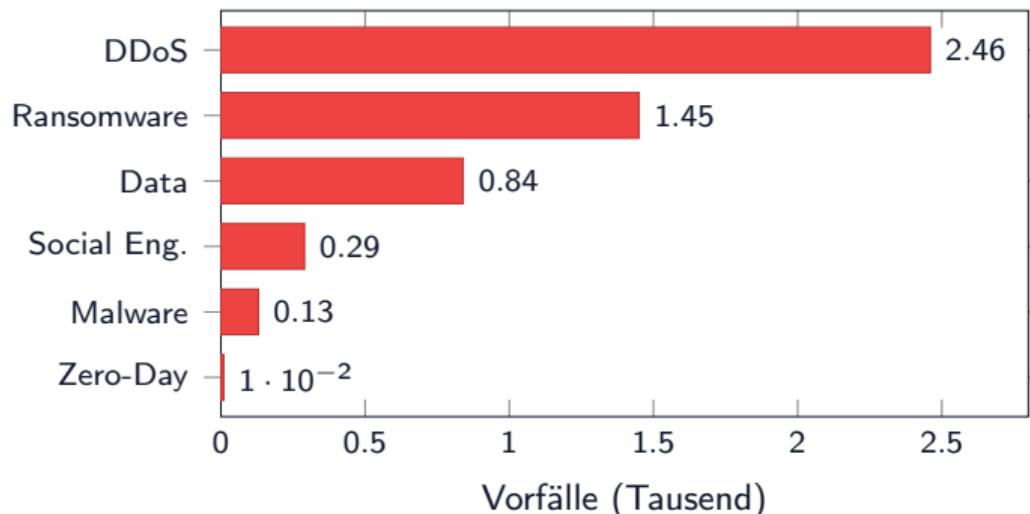
2024: 42 Mrd. USD

+740% Anstieg in 7 Jahren



Top-Bedrohungen EU:

1. DDoS/RDoS (46,31%)
2. Ransomware (27,33%)
3. Data Breaches (15,87%)
4. Social Engineering (5,37%)
5. Malware (2,45%)
6. Zero-Day (0,11%)



Ransomware: Zweitgrößte Bedrohung

Ziele & Motivation

Motivation der Angreifer

Warum diese Ziele?

Finanzielle Motive

Zahlungsbereitschaft bei kritischen Systemen

Sensible Daten

Erpressungspotenzial durch Datenleaks

Kritische Infrastrukturen

Hoher Druck durch Ausfallkosten

Geopolitische Faktoren

Destabilisierung und Spionage

Zunehmende Professionalisierung und Organisation

Warum die Bedrohung bleibt: Die Ökonomie

Ransomware-as-a-Service (RaaS) ist eine hochprofessionelle Schattenwirtschaft mit Milliardenumsätzen:

1. Access Provider

Spezialisten, die Zugänge zu Netzwerken potenzieller Opfer verkaufen.

2. RaaS Provider

Entwickler der Malware. Bieten C2-Infrastruktur und Support gegen Provision.

3. Affiliates

Kaufen Zugänge, mieten Malware und führen den Angriff durch.

Professionalisierungsgrad: Beispiel LockBit

Die Qualität der Services übertrifft teilweise legale SaaS-Anbieter.

Beispiel: LockBit 3.0 Bug-Bounty

LockBit hat ein eigenes Prämienprogramm für Hinweise zur Verbesserung ihrer Schadsoftware aufgelegt:

"Locker Bugs: Any errors during encryption ... that lead to corrupted files or to the possibility of decrypting files without getting a decryptor."

Wirtschaftliches Risiko

Im Gegensatz zu Banken ("Too big to fail") gilt für Ransomware-Gruppen: "**Too big to prevail**". Wer zu groß wird, zieht zu viel Aufmerksamkeit der Strafverfolger auf sich.

Arten von Ransomware

Grundlegende Typen

1. Crypto-Ransomware

- Verschlüsselt Benutzerdateien
- Verwendet AES, RSA
- System bleibt funktional
- Wiederherstellung oft unmöglich

2. Locker-Ransomware

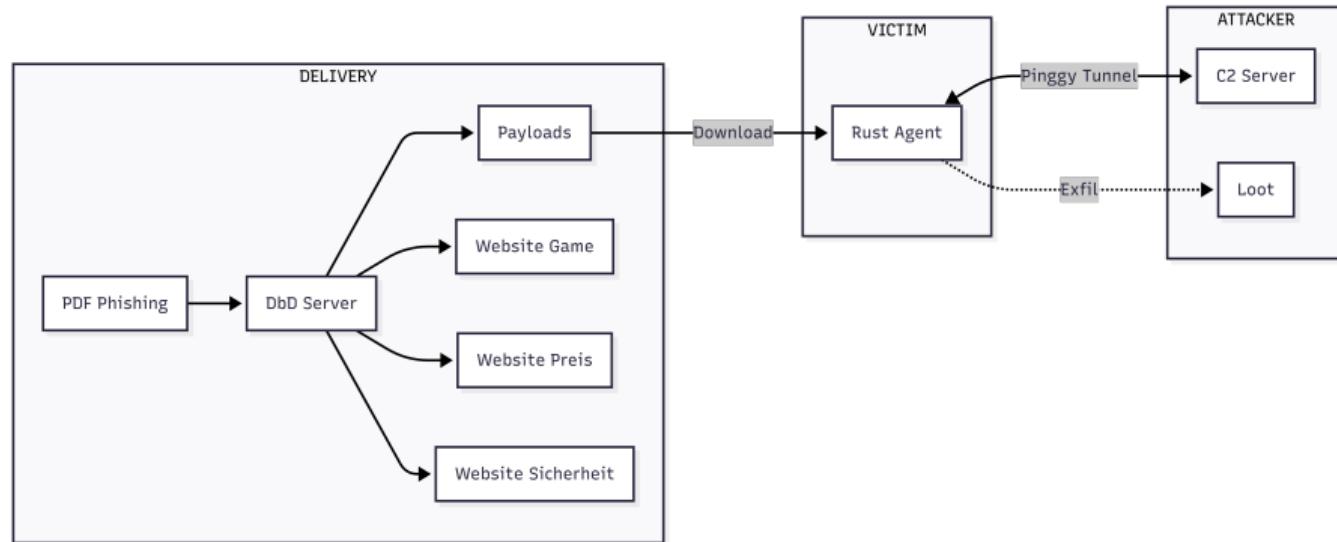
- Sperrt Systemzugriff
- Bildschirmsperre
- Keine Datenverschlüsselung
- Einfacher zu beheben

Projekt-Einblick: Edu-Ransomware

Unsere Ransomware ist eine klassische **Crypto-Ransomware mit Double Extortion**.

Case Study: Edu-Ransomware

Projektübersicht



Technische Architektur

Systemarchitektur & Technologie-Stack

Modulare Client–Server-Architektur (Reverse TCP Shell)

Agent (Client)	Rust	C2-Server (Control)	Python
<ul style="list-style-type: none">• Deployment: Statische Binary (Windows .exe, Linux .deb)• Core-Komponenten:<ul style="list-style-type: none">• main.rs – Initialisierung, Daemonisierung• evasion.rs – Ressourcen- & Zeitmanipulationsprüfung• crypto.rs – AES-256-CTR, atomare Dateioperationen• network.rs – Raw TCP, Retry-Logik, Protokollparser• extortion.rs – UI-Manipulation (Wallpaper, Browser)		<ul style="list-style-type: none">• Multithreaded TCP-Server• Sitzungsverwaltung paralleler Clients• Datenaufnahme (Base64-Streams → loot/)• Interaktive CLI	
<p>Delivery Infrastructure Bash / Python</p> <ul style="list-style-type: none">• User-Agent-basierte Payload-Auslieferung• Automatisierter Build- & Konfigurationsprozess			

Die Phasen eines Angriffs

Ablauf der Schadsoftware (Code-Logik):

1. Entry Point (`main.rs`)

- Initialer Start des Programms auf dem Zielsystem.

2. Sandbox Check (`evasion.rs`)

- **Erfolg (OK)**: Malware erkennt keine Analyse-Umgebung → Fortfahren.
- **Fehlschlag (FAIL)**: Malware beendet sich sofort (**Exit**), um Entdeckung zu vermeiden.

3. Persistence (`persistence.rs`)

- Einrichten eines **Autostarts**, damit die Malware Neustarts überdauert.

4. Command & Control (`network.rs`)

- Aufbau des **C2 Loops** zur Kommunikation mit dem Angreifer-Server.

5. Verschlüsselung (`crypto.rs`)

- Lokale Dateiverschlüsselung mittels **AES-256**.

6. Erpressung (`extortion.rs`)

- Anzeige der **Ransom Note** (Erpresserschreiben) für das Opfer.

Phase 1: Distribution & Infection

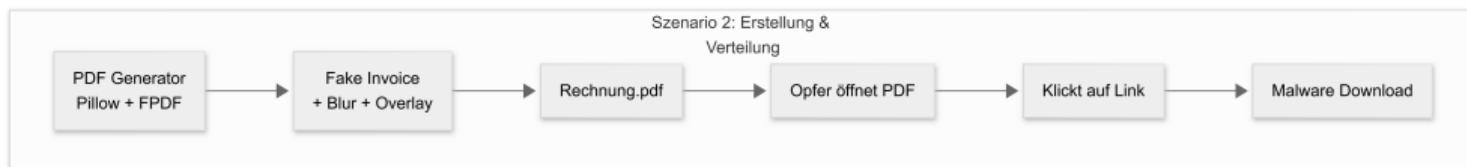
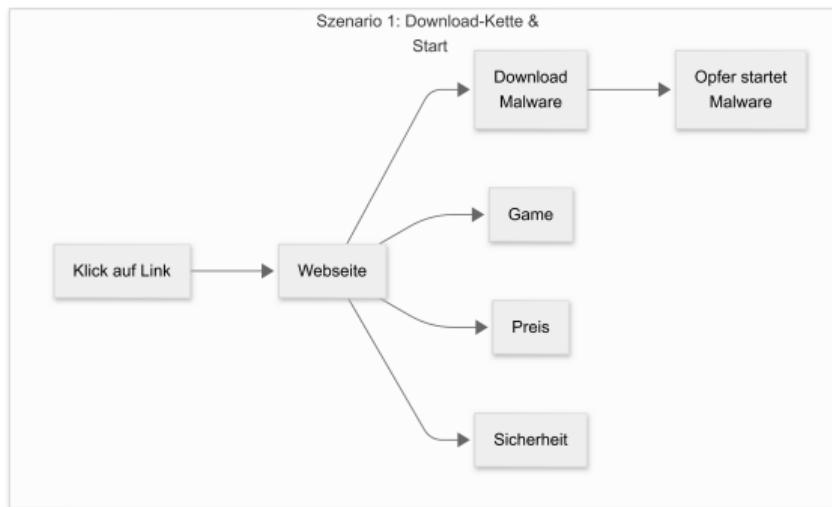
Theorie: Phishing (E-Mail Anhänge, Links), Drive-by-Downloads

Live Demo: Initial Access Szenario

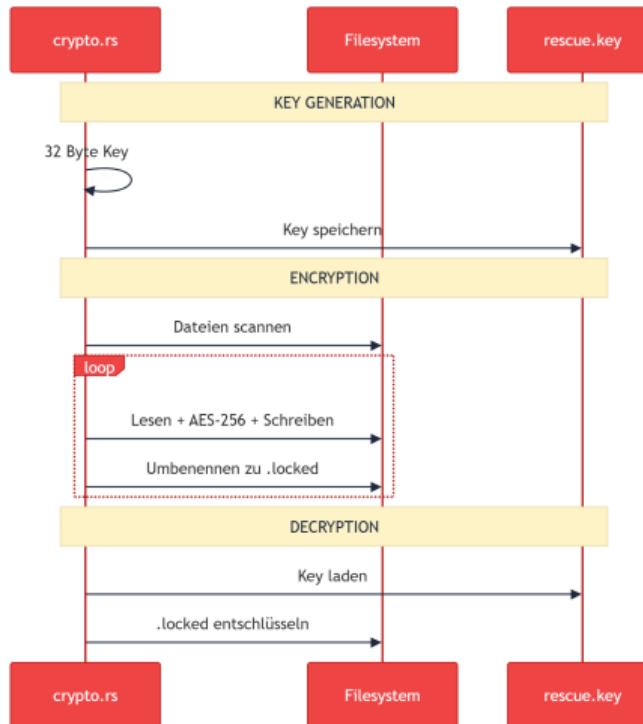
Szenario: Mitarbeiter erhält E-Mail mit "Rechnung_Dez.pdf"

- **Köder:** PDF enthält unscharfes Bild einer Rechnung
- **Trick:** Button "Inhalt entschlüsseln" (Social Engineering)
- **Ergebnis:** Smart-Server liefert Malware aus

Phase 1: Drive-by-Download Ablauf



Phase 1: PDF Phishing Technik



Phase 2: Execution & Evasion

Theorie: Ransomware installiert sich, prüft auf Sandboxes, etabliert Persistenz.

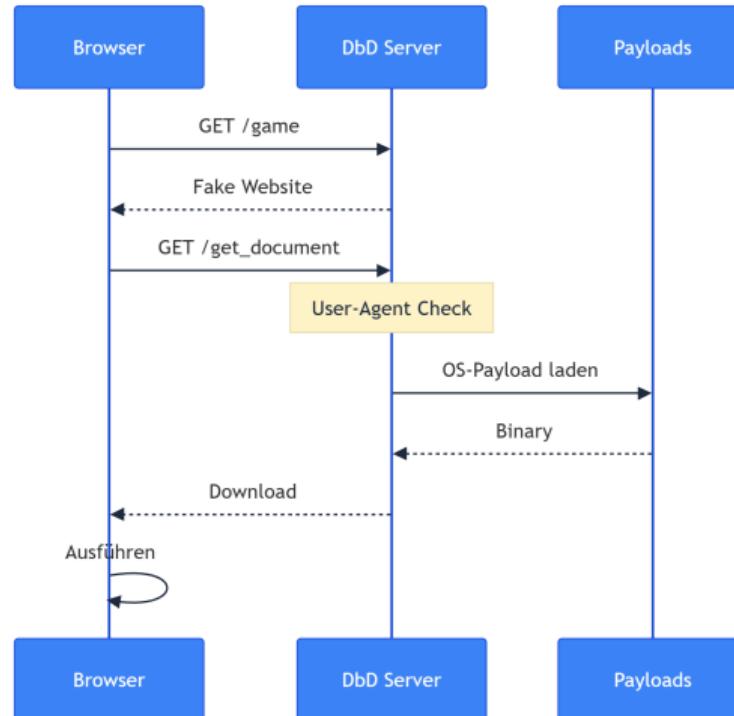
Projekt-Einblick: Evasion Module

Der Agent prüft beim Start:

- Ist RAM < 3GB?
- Sind weniger als 2 CPU-Kerne verfügbar?
- Sind weniger als 60GB Festplattenspeicher verfügbar?

Falls ja: **Sofortiger Abbruch** mit gefälschter Fehlermeldung.

Phase 2: Agent Architektur



Phase 3: C2 & Exfiltration

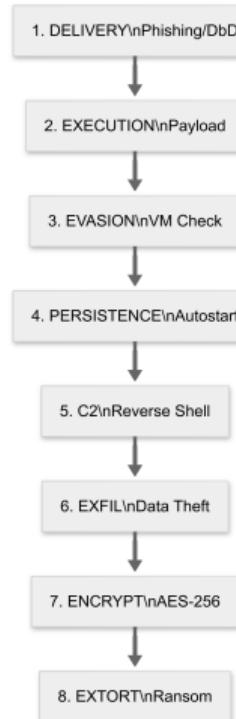
Theorie: Aufbau der Kommunikation, Nachladen von Befehlen, Datendiebstahl.

Live Demo: Attacker Control

Wir wechseln zum Angreifer-Terminal (C2):

- [+] New Victim Connected: ID 1
- Angreifer nutzt `shell`-Befehle zur Erkundung
- **Double Extortion:** `exfil secret.pdf` stiehlt Daten

Phase 3: C2 Kommunikation



Phase 4: Encryption (Impact)

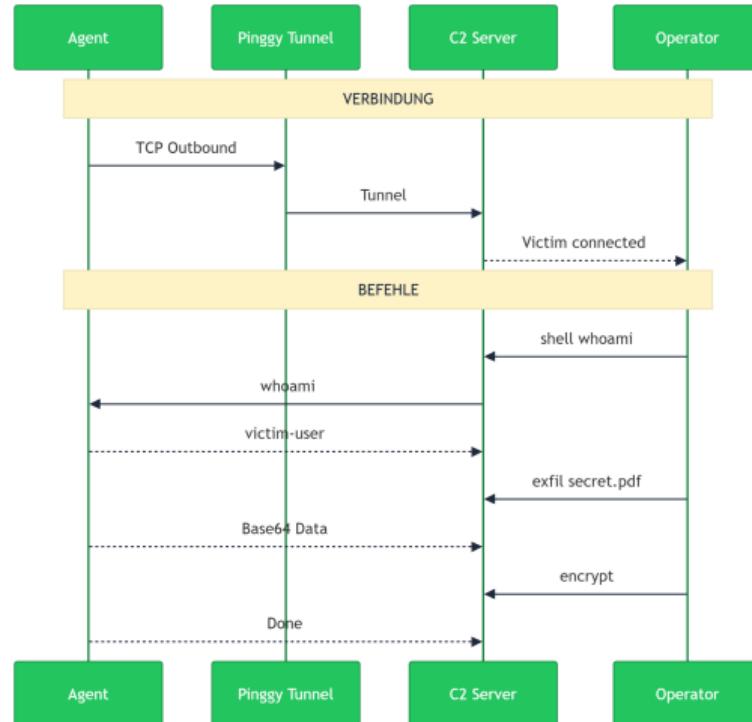
Theorie: Starke Verschlüsselung, Löschung von Backups.

Live Demo: The Panic Mode

Angreifer sendet `encrypt`. Auswirkungen auf Opfer-PC:

- Dateien erhalten Endung `.locked`
- Browser öffnet Lösegeldforderung (Stress)
- Log-File zeigt Verschlüsselung in Echtzeit

Phase 4: Verschlüsselungsprozess



Phase 5: Decryption (Recovery)

Szenario: Das Lösegeld wurde gezahlt (in der Simulation).

- Angreifer sendet Befehl `decrypt`
- Agent nutzt symmetrischen Key (AES-CTR)
- `.locked` Dateien verschwinden, Originale sind wieder da

Gegenmaßnahmen

Technische Maßnahmen:

Backups

3-2-1-Regel: 3 Kopien, 2 Medien, 1 Offsite

EDR / Antivirus

Verhaltensanalyse in Echtzeit

Projekt-Reflektion: Warum Evasion?

Herkömmliche AV-Systeme scannen oft statisch. Unsere Ransomware umgeht dies durch:

- Dynamisches Nachladen (keine Signatur beim PDF-Download)
- Sandbox-Checks (verhindert Cloud-Analyse)
- Nutzung von Rust (schwerer zu analysieren)

Detektionsansätze:

- **Statische Analyse:** Ohne Ausführung
- **Dynamische Analyse:** Sandbox
- **Machine Learning:** Unbekannte Varianten
- **Netzwerk:** Anomalie-Erkennung

Wichtige Merkmale:

- API-Aufrufe und Systemverhalten
- Datei-/Verzeichnisaktivitäten
- Netzwerkverkehrsmuster
- Verschlüsselungsoperationen

Sofortmaßnahmen bei Verdacht:

1. **Isolation** betroffener Systeme
2. Identifikation des Ransomware-Stamms
3. Bewertung der Schadenausbreitung
4. Benachrichtigung relevanter Stellen

Wiederherstellung:

- Aus Backups (wenn möglich)
- Key-Escrow-Mechanismen
- Forensische Analyse
- Systemhärtung vor Neustart

Keine Lösegeldzahlung (keine Garantie, finanziert weitere Angriffe)

Fazit

Zusammenfassung

Erkenntnisse aus der Projektentwicklung:

Komplexität

Als Anfänger von Null funktionsfähige Malware entwickelt – erschreckend niedrige Einstiegshürde.

KI-Unterstützung

Entwicklung mit KI-Tools beschleunigt den Prozess erheblich.

AV-Evasion

Malware-Scanner schlägt nicht an – Agent bleibt unerkannt.

Fazit

Unser Ziel war es herauszufinden, wie komplex die Implementierung funktionsfähiger Ransomware für Einsteiger ist. Das Ergebnis: **Mit modernen Tools und KI-Unterstützung ist die Hürde erschreckend niedrig.** Selbst gängige AV-Lösungen erkennen unseren Agent nicht.

Technologische Entwicklungen:

- **KI in Ransomware:** Adaptives Verhalten
- **Quantencomputing:** Bedrohung für Verschlüsselung
- **IoT-Ransomware:** Neue Angriffsflächen
- **Cloud-native:** Angriffe auf Cloud-Infra

Forschungsbedarf:

- Echtzeit-Schutz, Zero-Day-Erkennung
- Post-Quantum-Kryptographie
- Automatisierte Incident Response
- Internationale Strafverfolgung

Quellen

- [1] IT-Security Vorlesung, Prof. Dr. Matthias Hamann, WS24/25.
- [2] *The Age of Ransomware: A Survey on the Evolution, Taxonomy, and Research Directions* (Razulla et al.)
- [3] ENISA (2024): *Threat Landscape Report 2024*
- [4] *Edu-Ransomware Repository* (GitHub, 2026) – Eigene Entwicklung
- [5] Morato, D.; Berrueta, E.; Magaña, E.; Izal, M.: *Ransomware early detection by the analysis of file sharing traffic.*
- [6] Oz, H.; Aris, A.; Levi, A.; Uluagac, A. S.: *A Survey on Ransomware: Evolution, Taxonomy, and Defense Solutions.*