

VNC-Lasttest

Prof. Dr. rer. nat. Peter Gerwinski

5. April 2020

VNC-Lasttest

Prof. Dr. rer. nat. Peter Gerwinski

rerum naturalium = der natürlichen Dinge (lat.)

5. April 2020

VNC-Lasttest

- 1 Online-Werkzeuge und Privatsphäre**
 - 2 Virtual Network Computing (VNC)**
 - 3 Lasttests für VNC**
 - 4 Nachtisch: Visualisierung schwarzer Löcher**
- Weitere Informationen:
<https://gitlab.cvh-server.de/pgerwinski/ow>
<https://gitlab.cvh-server.de/pgerwinski/nit>
 - Direkt-Link zum Herunterladen dieser Vortragsfolien:
<https://gitlab.cvh-server.de/pgerwinski/nit/-/raw/master/20200405/vnc-20200405.pdf>

Stand: 5. April 2020

Copyright © 2020 Peter Gerwinski

Lizenz: CC-by-sa (Version 3.0) oder GNU GPL (Version 3 oder höher)

Sie können diese Vortragsfolien einschließlich L^AT_EX-Quelltext herunterladen unter:

<https://gitlab.cvh-server.de/pgerwinski/nit/-/tree/master/20200405>

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen
- bekannte Lösungen: leicht zu bedienen, stabil, oft kostenlos

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen
- bekannte Lösungen: leicht zu bedienen, stabil, oft kostenlos
- **Problem:** Stattdessen bezahlt man mit seinen Daten!

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen
- bekannte Lösungen: leicht zu bedienen, stabil, oft kostenlos
- **Problem:** Stattdessen bezahlt man mit seinen Daten!

Datenschutz und Sicherheit → Mehraufwand unvermeidbar

- Software, die Sie ausspioniert, ist leichter zu installieren, weil sie mehr über Sie weiß.
- Kommunikation über zentrale Server → kein eigener Wartungsaufwand, aber Anbieter kann mitlesen und manipulieren
- Anwendungs-Software hat gleichzeitig Verbindung zu ihrem Anbieter und Zugriff auf Kamera, Mikrofon und persönliche Daten
→ Fernwartung möglich, aber auch Fernsteuerung und Fernzugriff

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen
- bekannte Lösungen: leicht zu bedienen, stabil, oft kostenlos
- **Problem:** Stattdessen bezahlt man mit seinen Daten!

Datenschutz und Sicherheit → Mehraufwand unvermeidbar

- Software, die Sie ausspioniert, ist leichter zu installieren, weil sie mehr über Sie weiß.
- Kommunikation über zentrale Server → kein eigener Wartungsaufwand, aber Anbieter kann mitlesen und manipulieren
- Anwendungs-Software hat gleichzeitig Verbindung zu ihrem Anbieter und Zugriff auf Kamera, Mikrofon und persönliche Daten
→ Fernwartung möglich, aber auch Fernsteuerung und Fernzugriff

→ **Forderung:** dezentrale Server, 100 % Transparenz der Software

1 Online-Werkzeuge und Privatsphäre

- aktuell: großer Bedarf an Online-Werkzeugen
- bekannte Lösungen: leicht zu bedienen, stabil, oft kostenlos
- **Problem:** Stattdessen bezahlt man mit seinen Daten!

Datenschutz und Sicherheit → Mehraufwand unvermeidbar

- Software, die Sie ausspioniert, ist leichter zu installieren, weil sie mehr über Sie weiß.
- Kommunikation über zentrale Server → kein eigener Wartungsaufwand, aber Anbieter kann mitlesen und manipulieren
- Anwendungs-Software hat gleichzeitig Verbindung zu ihrem Anbieter und Zugriff auf Kamera, Mikrofon und persönliche Daten
→ Fernwartung möglich, aber auch Fernsteuerung und Fernzugriff

→ **Forderung:** dezentrale Server, 100 % Transparenz der Software

→ *freie Software (Open Source)*

1 Online-Werkzeuge und Privatsphäre

1.1 Freie Online-Werkzeuge für *Home Office* und Lehre

- [Mumble](#) für Audio-Konferenzen
- [BigBlueButton](#) und [OpenMeetings](#) als allgemeine Werkzeuge für Online-Präsentationen
- [noVNC](#) als Web-Interface für mit [VNC](#) exportierte Bildschirminhalte
- [Jitsi Meet](#) für Web-Konferenzen
- [OwnCloud](#) und [NextCloud](#) für den Austausch von Dateien
- [GitLab](#) als Web-Interface für [Git](#)-Repositories
- ... und natürlich die „klassischen“ Online-Werkzeuge:
SSH, IRC, screen/tmux, ytalk, ...

1 Online-Werkzeuge und Privatsphäre

1.2 Spezialisierte vs. Allzweck-Werkzeuge

- Allzweck-Werkzeuge:
[BigBlueButton](#), [OpenMeetings](#), [Jitsi Meet](#)
→ leichter Einstieg, schnelle Erfolge
- spezialisierte Werkzeuge:
[Mumble](#), [VNC](#), [noVNC](#), [OwnCloud](#), [NextCloud](#), [Git](#), [GitLab](#)
→ höhere Qualität und Stabilität, weniger Ressourcenverbrauch

1 Online-Werkzeuge und Privatsphäre

1.2 Spezialisierte vs. Allzweck-Werkzeuge

- Allzweck-Werkzeuge:
BigBlueButton, OpenMeetings, Jitsi Meet
→ leichter Einstieg, schnelle Erfolge
- spezialisierte Werkzeuge:
Mumble, VNC, noVNC, OwnCloud, NextCloud, Git, GitLab
→ höhere Qualität und Stabilität, weniger Ressourcenverbrauch

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten
- Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten
- Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten
- Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung
- Web-Interface durch darauf spezialisierte Software realisiert (noVNC)

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten
- Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung
- Web-Interface durch darauf spezialisierte Software realisiert (noVNC)
→ *machte letzte Woche Probleme*

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
- um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
- zahlreiche Anbieter
- spezialisiert auf Übertragung von Bildschirmhalten
- Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung
- Web-Interface durch darauf spezialisierte Software realisiert (noVNC)
→ *machte letzte Woche Probleme*

→ durch Analyse des Quelltextes von noVNC gefunden:
fehlerhafte Komponente (websockify)

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
 - um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
 - zahlreiche Anbieter
 - spezialisiert auf Übertragung von Bildschirmhalten
 - Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung
 - Web-Interface durch darauf spezialisierte Software realisiert (noVNC)
→ *machte letzte Woche Probleme*
- durch Analyse des Quelltextes von noVNC gefunden:
fehlerhafte Komponente (websockify)
- Lösung: Austausch dieser Komponente durch ein Konkurrenzprodukt
(ebenfalls freie Software)

2 Virtual Network Computing (VNC)

- Fernwartungs-Software
 - um 1990 entwickelt, seit 1998 freie Software (GNU GPL)
 - zahlreiche Anbieter
 - spezialisiert auf Übertragung von Bildschirmhalten
 - Sicherheit durch andere, darauf spezialisierte Software realisiert (SSH)
→ sicherer als eingebaute Sicherheitslösung
 - Web-Interface durch darauf spezialisierte Software realisiert (noVNC)
→ *machte letzte Woche Probleme*
- durch Analyse des Quelltextes von noVNC gefunden:
fehlerhafte Komponente (websockify)
- Lösung: Austausch dieser Komponente durch ein Konkurrenzprodukt
(ebenfalls freie Software)
- **Bei nicht-freier Software: nur Anbieter kann Fehler beheben**

3 Lasttests für VNC

- Vorlesung mit Folien
- Uhr mit Sekundenzeiger
- Molpy *Up!*
- Malen mit GIMP
- Kamerabild
- (Simulierte? ;–) Netzwerk-Probleme auf Vortragendenseite

4 **Nachtisch: Visualisierung schwarzer Löcher**

4 Nachtisch: Visualisierung schwarzer Löcher



Quelle: [Event Horizon Telescope \(EHT\)](#)
Verwendung gemäß Lizenz: [CC BY 4.0](#)

4 Nachtisch: Visualisierung schwarzer Löcher

4.1 Was ist ein schwarzes Loch?

- Allgemeine Relativitätstheorie
- Raum und Zeit gehören zusammen: Raumzeit
- Gravitation = Krümmung der Raumzeit
- schwarzes Loch: so starke Krümmung, daß Licht und andere Informationen einen Bereich nicht verlassen können (Ereignishorizont).
- Radius dieses Bereichs: Schwarzschild-Radius
- in der Mitte: punktförmige Singularität

4 **Nachtisch: Visualisierung schwarzer Löcher**

4.2 **Was passiert, wenn man in das schwarze Loch fällt?**

Beim Passieren des Ereignishorizonts wechseln Raum und Zeit ihre Bedeutung:

- Was vorher die Zeit war, ist jetzt eine normale Raumdimension, in der man vorwärts und rückwärts reisen kann.
- Die Raumrichtung hin zum Mittelpunkt wird zur neuen Zeit.

4 Nachtisch: Visualisierung schwarzer Löcher

4.2 Was passiert, wenn man in das schwarze Loch fällt?

Beim Passieren des Ereignishorizonts wechseln Raum und Zeit ihre Bedeutung:

- Was vorher die Zeit war, ist jetzt eine normale Raumdimension, in der man vorwärts und rückwärts reisen kann.
- Die Raumrichtung hin zum Mittelpunkt wird zur neuen Zeit.

—→ Das schwarze Loch wieder zu verlassen, entspräche einer Reise in die Vergangenheit.

4 Nachtisch: Visualisierung schwarzer Löcher

4.2 Was passiert, wenn man in das schwarze Loch fällt?

Beim Passieren des Ereignishorizonts wechseln Raum und Zeit ihre Bedeutung:

- Was vorher die Zeit war, ist jetzt eine normale Raumdimension, in der man vorwärts und rückwärts reisen kann.
- Die Raumrichtung hin zum Mittelpunkt wird zur neuen Zeit.

—→ Das schwarze Loch wieder zu verlassen, entspräche einer Reise in die Vergangenheit.

Beim Erreichen des Mittelpunkts hören Raum und Zeit auf.

4 Nachtsch: Visualisierung schwarzer Löcher

4.3 Wie kann man sich ein schwarzes Loch vorstellen?

4 **Nachtisch: Visualisierung schwarzer Löcher**

4.3 **Wie kann man sich ein schwarzes Loch vorstellen?**

- Das ist nicht einfach. ;—)

4 Nachtisch: Visualisierung schwarzer Löcher

4.3 Wie kann man sich ein schwarzes Loch vorstellen?

- Das ist nicht einfach. ;—)
- Eine gekrümmte zweidimensionale Fläche kann man sich als Oberfläche eines dreidimensionalen Objekts vorstellen.
- Beispiel: Kugeloberfläche

4 Nachtisch: Visualisierung schwarzer Löcher

4.3 Wie kann man sich ein schwarzes Loch vorstellen?

- Das ist nicht einfach. ;—)
- Eine gekrümmte zweidimensionale Fläche kann man sich als Oberfläche eines dreidimensionalen Objekts vorstellen.
- Beispiel: Kugeloberfläche
- Eine gekrümmte vierdimensionale Raumzeit kann man sich als Oberfläche eines sechsdimensionalen Objekts vorstellen.
- Beispiel: Fronsdal-Einbettung eines schwarzen Lochs.
(Man kann beweisen, daß fünf Dimensionen nicht genügen.)

Literatur: <https://arxiv.org/pdf/1202.1204> und enthaltene Referenzen

4 Nachtisch: Visualisierung schwarzer Löcher

4.4 Visualisierungen der Fronsdal-Einbettung eines schwarzen Lochs

Fronsdal-Einbettung: gekrümmte vierdimensionale Raumzeit als Oberfläche eines sechsdimensionalen Objekts

- 3 von 6 Dimensionen:
2 Raum-Dimensionen, 1 Krümmungs-Dimension
- 3 von 6 Dimensionen:
1 Raum-Dimension, 1 Zeit-Dimension, 1 Krümmungs-Dimension
- 4 von 6 Dimensionen:
2 Raum-Dimensionen, 1 Zeit-Dimension, 1 Krümmungs-Dimension
(perspektivisches 3d-Bild eines 4d-Objekts)

Vielen Dank für Ihre Aufmerksamkeit! :-)

1 Online-Werkzeuge und Privatsphäre

2 Virtual Network Computing (VNC)

3 Lasttests für VNC

4 Nachtisch: Visualisierung schwarzer Löcher

- Weitere Informationen:

<https://gitlab.cvh-server.de/pgerwinski/ow>

<https://gitlab.cvh-server.de/pgerwinski/nit>

- Direkt-Link zum Herunterladen dieser Vortragsfolien:

<https://gitlab.cvh-server.de/pgerwinski/nit/-/raw/master/20200405/vnc-20200405.pdf>

Stand: 5. April 2020

Copyright © 2020 Peter Gerwinski

Lizenz: CC-by-sa (Version 3.0) oder GNU GPL (Version 3 oder höher)

Sie können diese Vortragsfolien einschließlich L^AT_EX-Quelltext herunterladen unter:

<https://gitlab.cvh-server.de/pgerwinski/nit/-/tree/master/20200405>