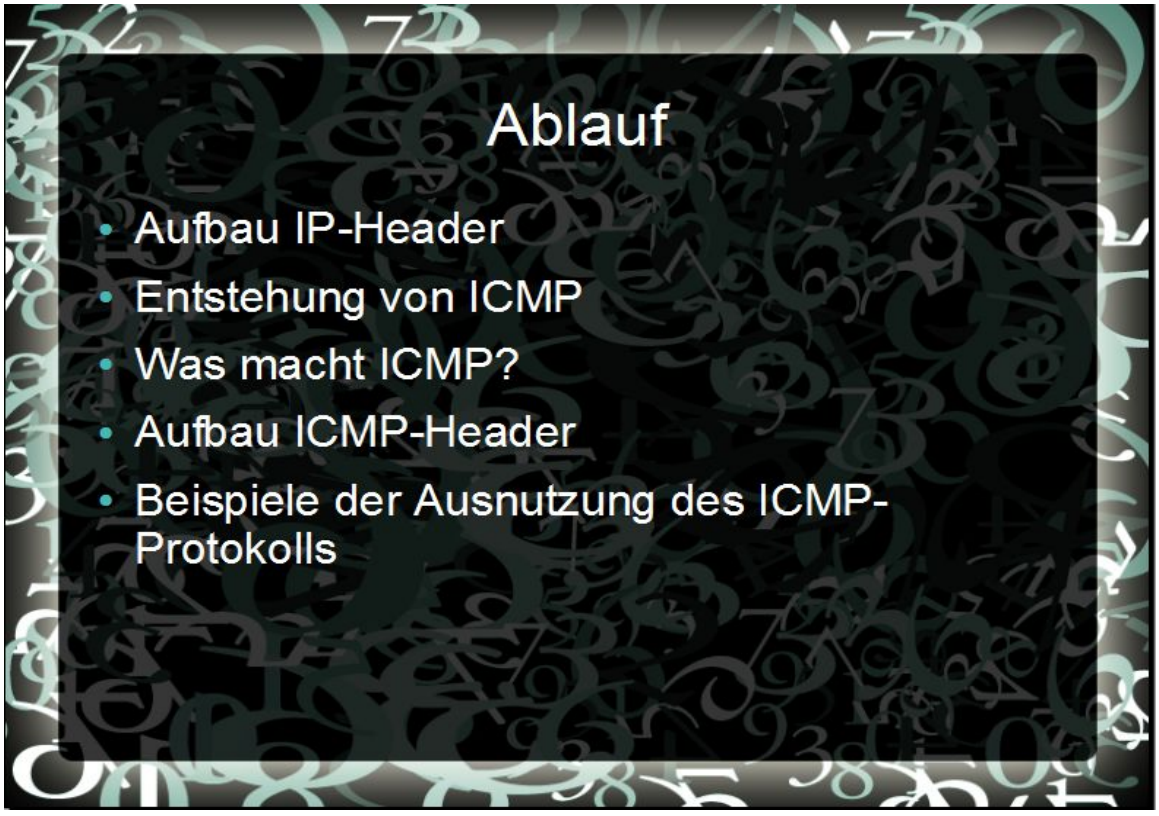
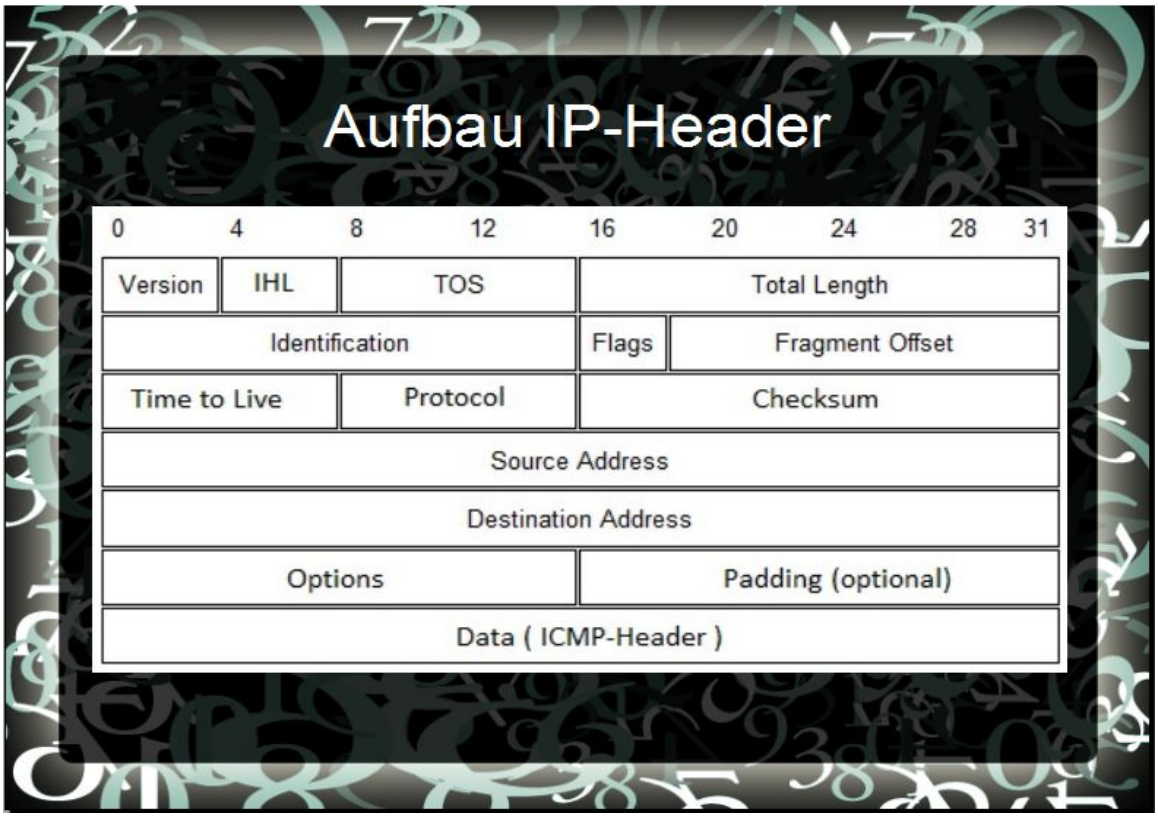
Folie 1

Folie 2



Folie 3



Version:

- Das Versionsfeld gibt an ob es sich um IPv4 oder um IPv6 handelt.

IHL (IP Header Length)

- Im IHL-Feld wird ein vielfaches von 32 Bit angegeben. Die Summe gibt die Größe des IP Headers an.

TOS (Type of Service)

- Das TOS-Feld wird für die Priorisierung des Datenpaketes genutzt.

Total Length

- Dieses Feld gibt die Gesamtgröße des Paketes an(IP-Header + Daten).

Identification, Flags, Fragment Offset

- Mit diesen 3 Feldern wird die Fragmentierung des Datenpaketes gesteuert.

TTL(Time to Live)

- Gibt die Lebensdauer des Pakets an. Der Wert wird von jedem Router um eins verringert. Hat das Feld den Wert null erreicht, wird das Paket verworfen.

Protocol

- Dieses Feld gibt an welches Protokoll sich im Datenbereich befindet.
  z.B. 1=ICMP, 6=TCP, 17=UDP

Checksum

- Prüfsumme des IP-Headers.

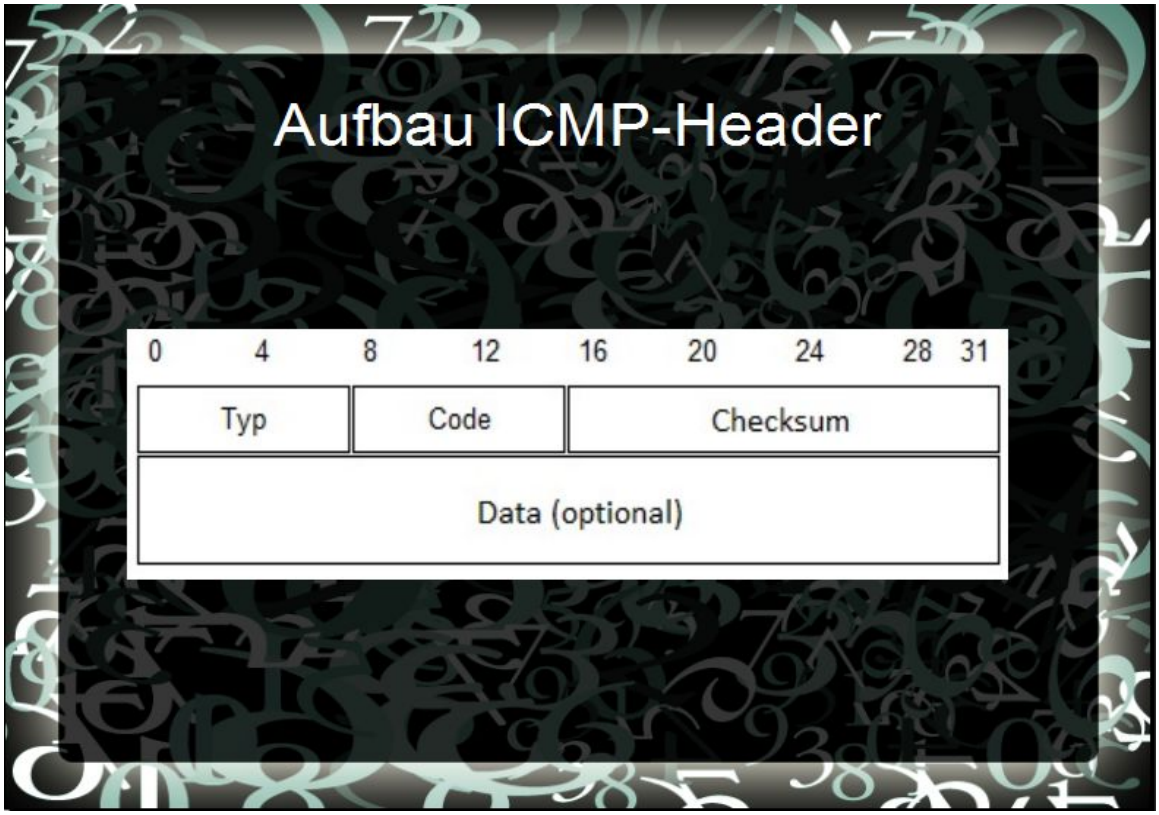
Folie 4

Bei einer wissenschaftlichen Konferenz im Jahre 1967, welche sich mit dem Aufbau eines einzigen Computernetzwerkes befasste, wurden auch die Grundlegenden Protokolle festgelegt. Zu diesen zählten unter anderem ICMP, TCP/IP und UDP.
Das Protokoll ICMP ist in der RFC 792 festgelegt.
Des Weiteren sind einige neue Erweiterungen in der RFC 1122 beschrieben.

Folie 5

Das Protokoll ICMP ist zuständig für die Information der beteiligten Stationen über Diagnose Informationen, Fehlermeldungen und Informationsmeldungen.

Folie 6



Type:

- Dieses Feld gibt den grundlegenden Fehler an.

Code:

- Dieses Feld spezifiziert den Fehler, soweit es eine Spezifizierung für diesen gibt.

Checksum

- Das Feld Checksum beinhaltet die Prüfsumme des ICMP-Headers.

Data(optional)

- Im Datenfeld wird in den meisten Fällen der IP-Header und 64 bit der Daten des auslösenden Paketes geschickt.

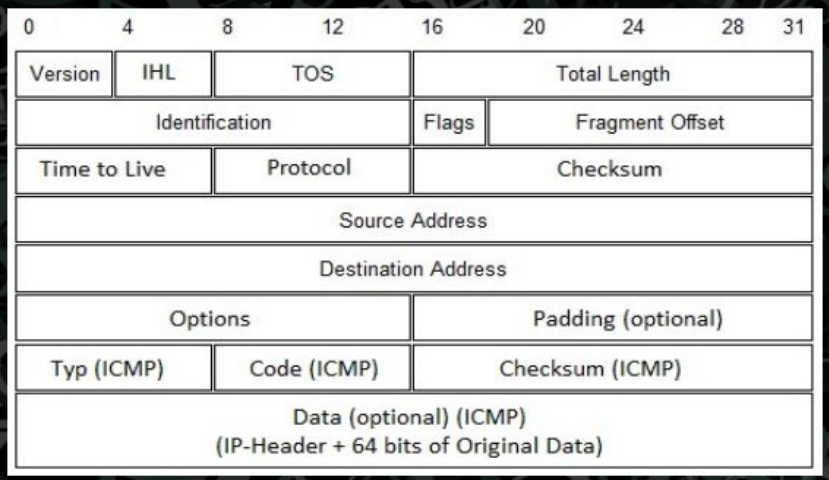
Folie 7

Aufbau ICMP-Header

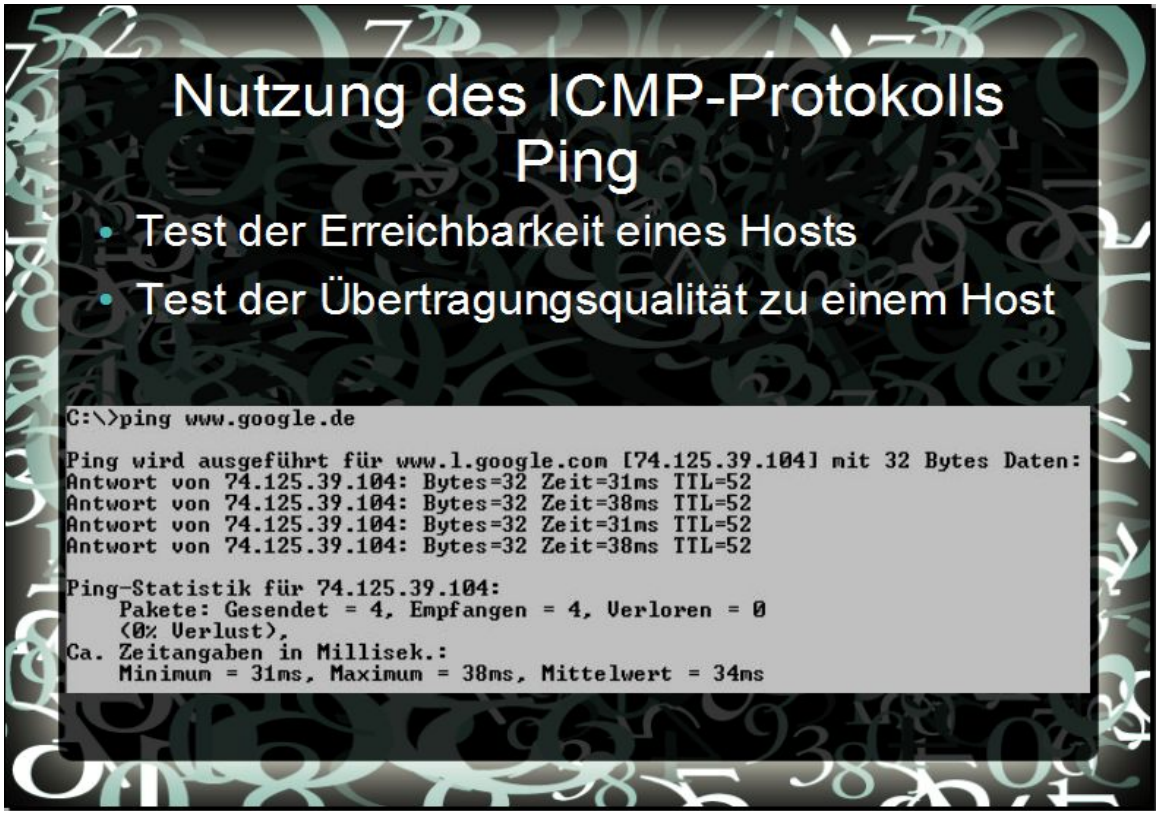
Typ	Typname	Code	Beschreibung
0	Echo Reply	0	Ping Answer
3	Destination Unreachable	0	Net Unreachable
		1	Host Unreachable
		2	Protocol Unreachable
		3	Port Unreachable
		4	Fragmentation Needed and Don't Fragment was Set
4	Source Quench	0	(Datagramm verworfen, Warteschlange voll)
8	Echo-Request	0	Ping Request
11	Time Exceeded	0	Time to Live exceeded in Transit
		1	Fragment Reassembly Time Exceeded

Folie 8

IP-Header samt ICMP

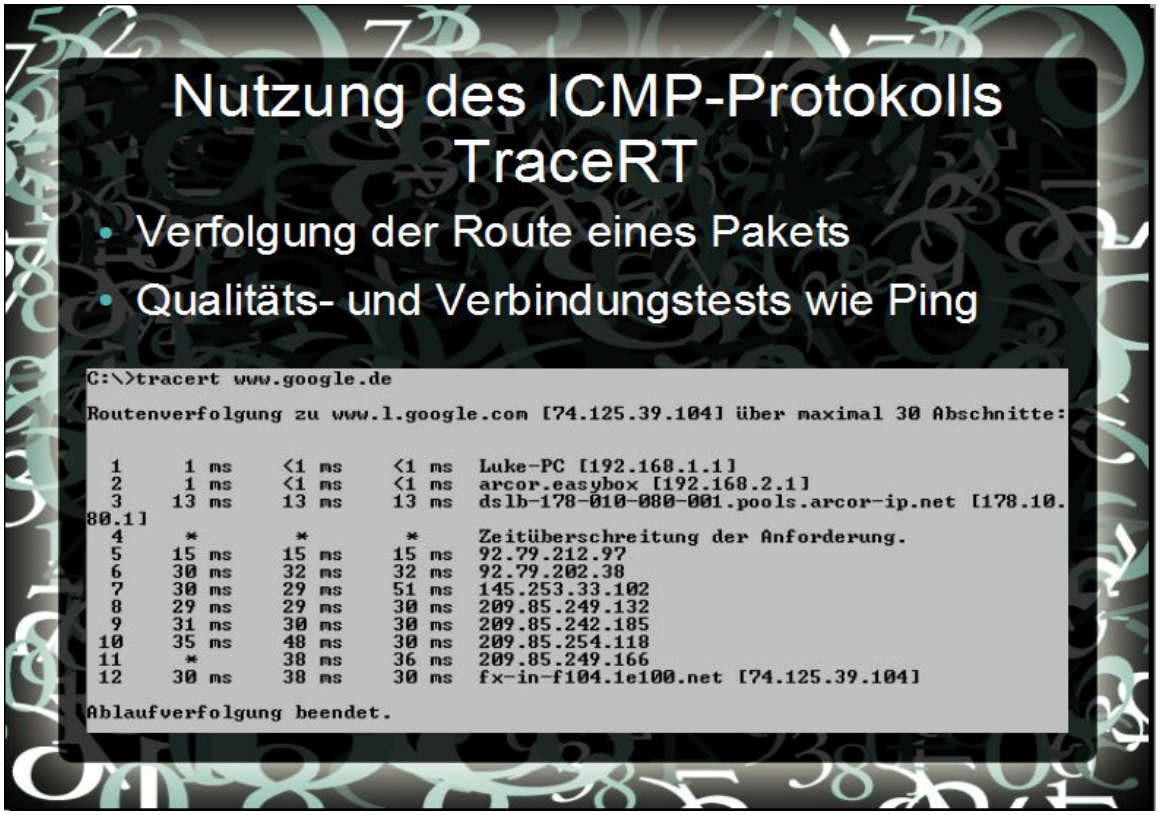


Folie 9



Das Programm Ping sendet ICMP-Echo-Request Pakete und wartet auf ein entsprechendes ICMP-Echo-Reply Paket. Dabei informiert Ping nicht nur darüber ob eine Verbindung zwischen zwei Stationen möglich ist, sondern auch über die Qualität der Verbindung.
Es zeigt in einer Übersicht die Anzahl der gesendeten Pakete, die Anzahl der empfangenen Pakete, die benötigte Zeit und die Anzahl der verlorenen Pakete an.

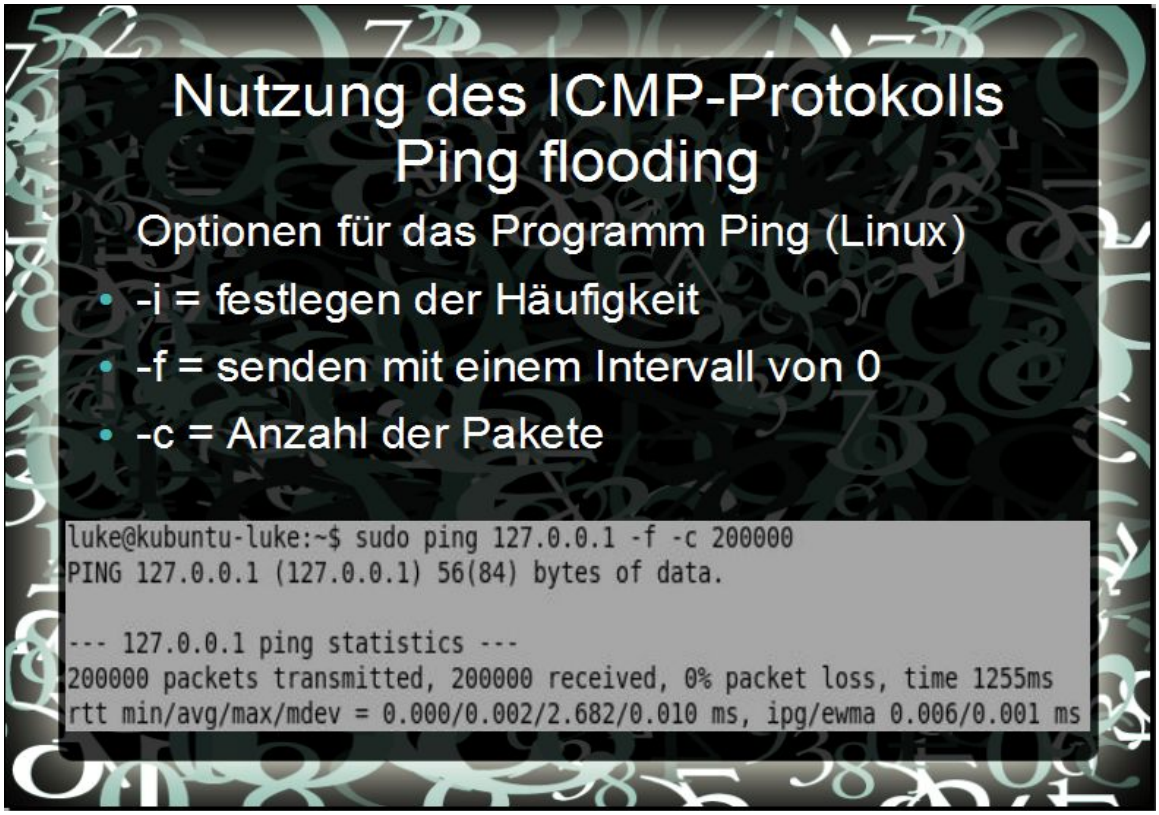
Folie 10



Das Programm TraceRoute arbeitet nach einem ähnlichen Prinzip wie Ping. Bei TraceRT wird aber der Wert des TTL Feldes auf eins gesetzt und mit jedem Paket um eins erhöht bis das Ziel erreicht ist. Somit erhält man von jeder Station ein ICMP Time to Live exceeded Paket zurück und kann mit diesen Informationen eine Liste der Sprünge bis zum Ziel erstellen.

Dieses Programm ist in meinen Augen der Weg zu einer nicht geplanten Nutzung dieses Protokolls.

Folie 11



Das Programm Ping hat noch einige Optionen, welche es leicht machen mit diesen einen schwachen Angriff auszuüben.

Die wichtigsten Optionen sind folgende:

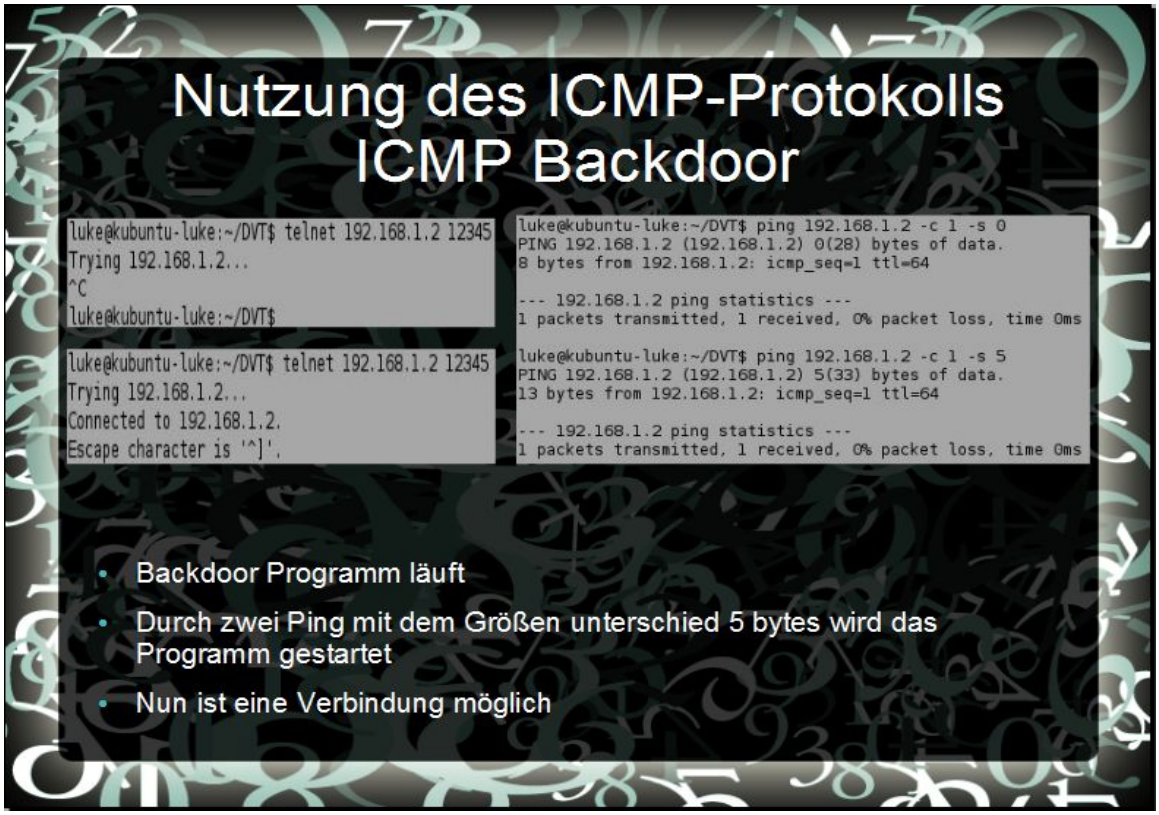
-i = Erlaubt die Festlegung der Häufigkeit in der Pakete versendet werden.

-c = Gibt die Anzahl der zu versendenden Pakete an.

-s = Erlaubt die Festlegung der Paket Größe.

-f = Mit dieser Option sendet das Programm Ping sofort nach Erhalt des Echo-Reply Paketes ein neues Echo-Request Paket.

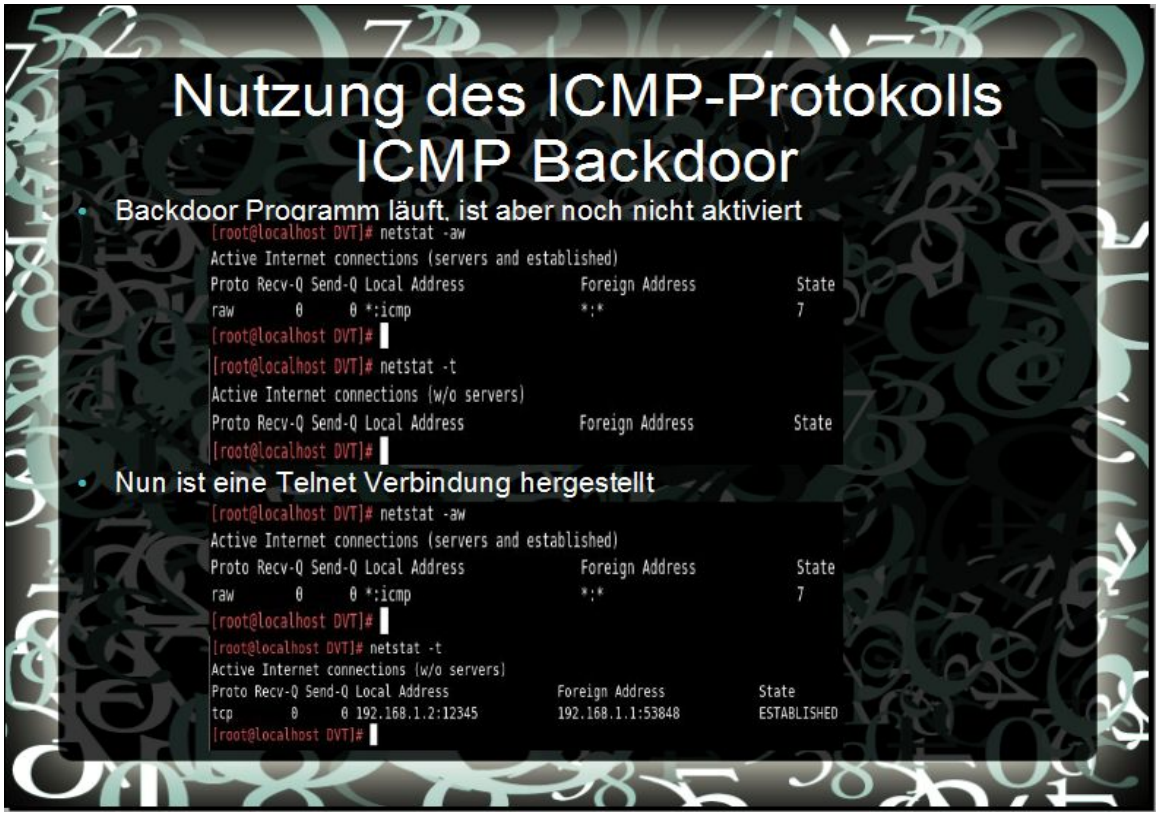
Folie 12



Es gibt die Möglichkeit ein Programm zu schreiben, welches die eintreffenden Ping Pakete auswertet und bei bestimmten Konstellationen eine vorgegebene Aufgabe ausführt.

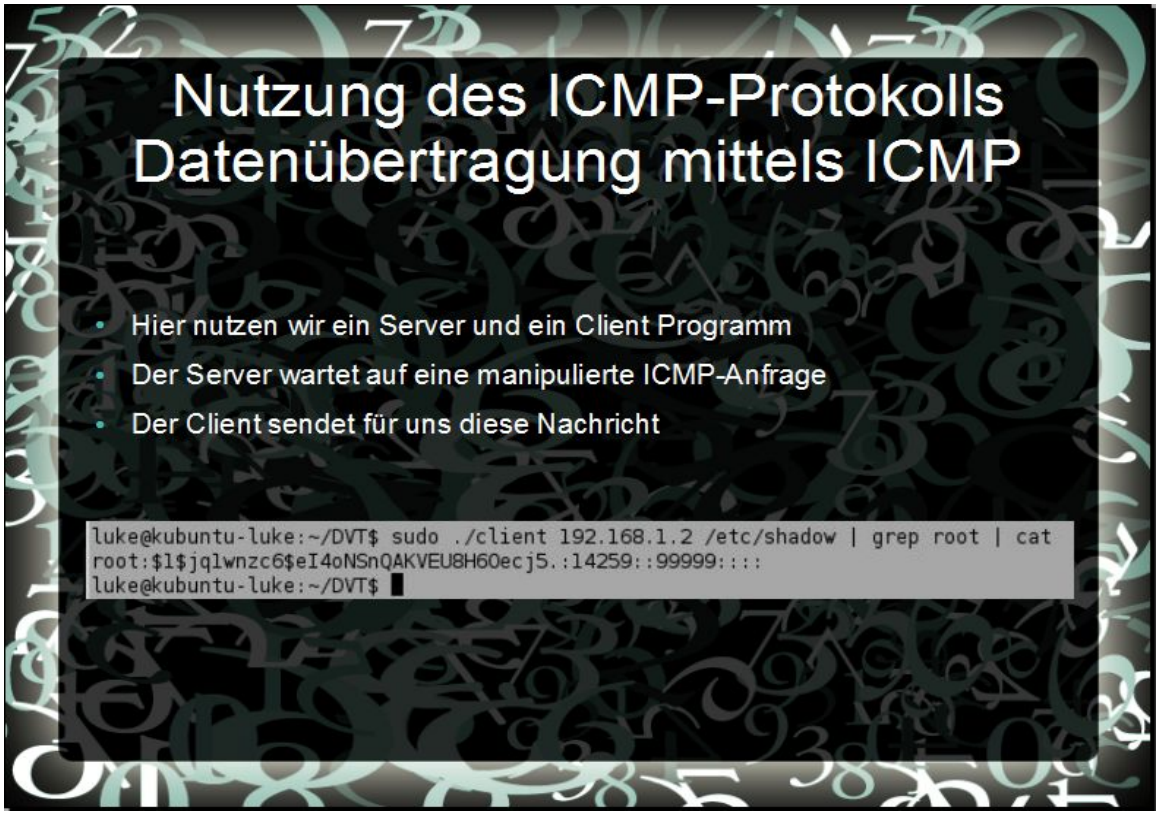
Das von mir in diesem Beispiel verwendete Programm wartet auf zwei aufeinander folgende Echo-Request Pakete mit einem Größenunterschied von 5 byte. Sind diese eingetroffen wird eine Remote-Konsole für eine Telnet Verbindung mit Root-Rechten gestartet.

Folie 13



Der Vorteil zu einem mit TCP/IP arbeitenden Programm ist, dass ICMP Sockets seltener Kontrolliert werden und wenig Rückschluss auf den Angreifer bieten. Leider ist bei diesem Beispiel zu sehen das zwar im wartenden Zustand nur ein ICMP-Socket angezeigt wird, aber sobald sich jemand über Telnet verbindet wieder eine offene TCP/IP Verbindung erscheint. Diese ist jedoch nur für die Zeit sichtbar, in der die Verbindung besteht.

Folie 14

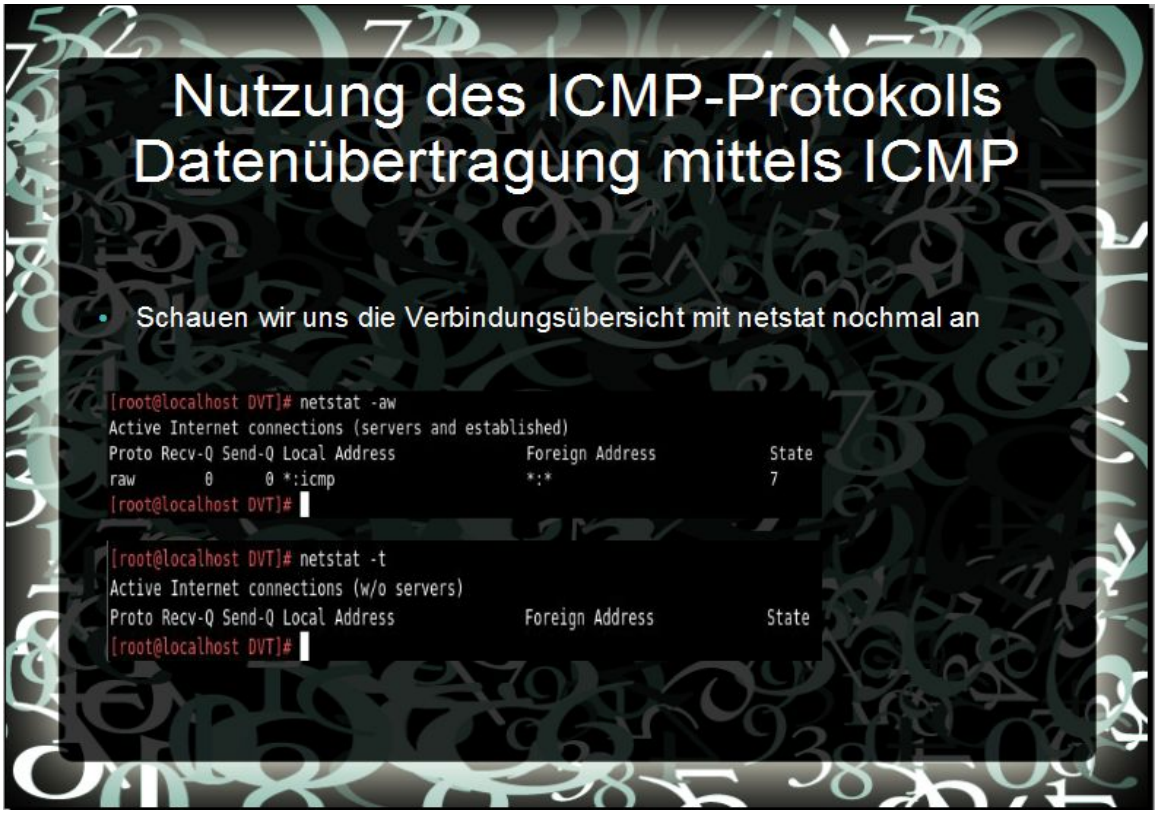


ICMP bietet auch eine Möglichkeit der Datenübertragung ohne TCP/IP, mit dem Vorteil, dass diese Verbindung von Betreuer nicht entdeckt wird.

Das hier in diesem Beispiel benutzte Programm sendet eine manipulierte ICMP Nachricht, in welcher der Inhalt einer bestimmten Datei angefordert wird.

Das auf dem angegriffenen PC laufende Programm wertet diese Nachricht aus und packt den Inhalt der angeforderten Datei in das Datenfeld des ICMP-Headers, welches eigentlich für die Übertragung des auslösenden IP-Headers und dem Anfang der Daten aus dem IP Paket gedacht ist. Der sinngemäße Grund für die Übertragung des IP-Headers und den ersten 64 bit der Daten aus selbigem, ist dem Sender die Möglichkeit zu geben, dass fehlerhaft übertragene Paket zu ermitteln und neu zu senden.

Folie 15



Wie wir sehen können besteht bei diesem Beispiel immer nur der ICMP-Socket, welcher auch von einem normalen Ping Programm sein könnte.

Folie 16

Folie 17