



Lehrstuhl Netzarchitekturen und Netzdienste  
Institut für Informatik  
TU München

# Wie schreibt man eine Ausarbeitung?

Prof. Dr.-Ing. Georg Carle  
Lehrstuhl für Netzarchitekturen und Netzdienste  
TU München



## Neue Templates!!!

- Auf der Proseminar-Homepage sind ab heute **neue Templates** verlinkt
- Bitte diese verwenden!!



# Übersicht

- Ziele
  - eines (Pro)Seminars
  - einer Ausarbeitung
  
- Struktur der Ausarbeitung
  
- Tipps und Richtlinien zur Ausarbeitung
  
- Tipps zum Vortrag



# Übersicht

- **Ziele**
  - **eines (Pro)Seminars**
  - **einer Ausarbeitung**
  
- Struktur der Ausarbeitung
  
- Tipps und Richtlinien zur Ausarbeitung
  
- Tipps zum Vortrag



# Ziele eines (Pro)Seminars

- Für einen selbst:
    - Üben wissenschaftlicher Arbeitsweise, insb. Schreiben eines Papers
    - Vorbereitung auf
      - BA
      - SEP
      - DA
      - MA
  
  - Für andere:
    - Einführung in ein interessantes sowie aktuelles (Forschungs)-Thema
    - Ausarbeitung kann auch zukünftigen Studenten zur Verfügung gestellt werden
- ➔ Ziel: „Jeder kennt das Thema von jedem“



## Ziele der Seminaarausarbeitung

- Kompakte Informationen über ein Themengebiet vermitteln
- Schwerpunkt liegt auf Darstellung von gegebenen Tatsachen, nicht auf Präsentation neuer eigener Erkenntnisse
- Zielpublikum sind andere Seminarteilnehmer, die das Thema nicht oder kaum kennen
  - Diese sollen ein Verständnis für eine für sie neue Technologie/ein neues Verfahren/Protokoll... entwickeln



# Übersicht

- Ziele
  - eines (Pro)Seminars
  - einer Ausarbeitung
  
- **Struktur der Ausarbeitung**
  
- Tipps und Richtlinien zur Ausarbeitung
  
- Tipps zum Vortrag



# Struktur der Ausarbeitung

- (... oder eines Papers)
  
- Typische Struktur einer Ausarbeitung:
  - Kurzfassung (Abstract)
  - Einführung (Introduction)
  - Rumpf
  - Verwandte Arbeiten (Related Work)
  - Zusammenfassung (Conclusion)
    - Schlussfolgerung
    - Evtl. Ausblick
  - Literaturreferenzen (References)





# Ein Beispiel...

## An End to the Middle

Colin Dixon

Arvind Krishnamurthy  
University of Washington

Thomas Anderson

### Abstract

The last fifteen years has seen a vast proliferation of middleboxes to solve all manner of persistent limitations in the Internet protocol suite. Examples include firewalls, NATs, load balancers, traffic shapers, deep packet intrusion detection, virtual private networks, network monitors, transparent web caches, content delivery networks, and the list goes on and on. However, most smaller networks in homes, small businesses and the developing world are left without this level of support. Further, the management burden and limitations of middleboxes are apparent even in enterprise networks.

We argue for a shift from using proprietary middlebox hardware as the dominant tool for managing networks toward using open software running on end hosts. We show that functionality that seemingly must be in the network, such as NATs and traffic prioritization, can be more cheaply, flexibly, and securely provided by distributed software running on end hosts, working in concert with vastly simplified physical network hardware.

### 1 Introduction

Middleboxes have done wonderful things for networks in the past few decades. They have enabled us to deal with address-space depletion (NATs), overcome TCP's limitations (packet shapers), and provide better security without changing commodity operating systems (firewalls). Beyond these we rely on virtual private networks (VPNs), proxies, caches, intrusion detection systems (IDSs), load balancers, and many other offerings to keep our networks working properly so we can accomplish our day-to-day tasks.

However, middleboxes have not solved everyone's problems. Most smaller networks have been left out in the cold as the costs to buy and run middleboxes are simply too high. While we could hope that commodity home routers will eventually include all the functionality of the middleboxes we use in enterprise networks, we'd rather not wait. A key motivation for the authors is to be able to get the level of IT support we have at work, for our own networks at home and for networks we help manage for non-profits in the developing world.

We aim to solve two key shortcomings of the current middlebox-based approach to network management in order to make it more effective and its benefits more accessible.

- Cost Middleboxes are expensive at almost every level:

upfront costs of hardware, skilled staff to manage them, long-term costs of vendor lock-in, scaling to handle increased loads, and so on. In contrast, most computers today are massively overprovisioned, with a proliferating number of cores used only for short bursts of user activity. It should seem strange then to intentionally design a system that keeps functionality in the network and away from endpoints.

- Perimeter-based Middleboxes are inherently point solutions, or at best perimeter solutions, yet today's LAN's have increasingly complex internal structure and the proliferation of mobile devices makes the very notion of inside and outside quaint. To get the full benefit of a middlebox, it needs to be deployed throughout the network, not just at the edge. While we can conceivably integrate the full set of middlebox functionality with every router and LAN switch, the result would be neither clean nor cheap.

In the face of these challenges, we propose a more radical, quicker and cheaper option. Specifically, let us consider if we need middleboxes at all. Can we return to a world where we have a simple network with intelligent endpoints and be better off for it? Knowing what we know today, if we were to start from scratch we would not architect non-interoperable complexity into network devices. Rather, we would ask what is the cleanest network we can design to provide us the services we need at reasonable cost?

Several technology trends make this approach increasingly feasible to consider. Perhaps foremost is trusted computing: cheap hardware to validate that a particular device is running a well-known piece of software. In the current model, network administrators operate without the aid of endpoints as they are considered untrusted components. Instead, trusted computing hardware can make it possible to move enforcement onto the endpoint without compromising security. (Of course, some provision must be made in the network for untrusted nodes, but these can be the rare case.) Virtual machines allow this trusted enforcement code to be isolated from the rest of the endpoint operating system; to the user, their computer appears just as configurable as it always has been. Finally, the transition to multicore architectures suggests that all this can be done without even much of a performance penalty, as the PC core running the network enforcement code would likely have been idle in any event.

We call our approach "End to the Middle" or ETTM—

the operating system and observe which applications are generating or requesting the traffic this task becomes more tractable. Even more advanced quality of service tasks such as providing a boost to traffic related to the currently active window are possible.

### 4 Related Work

A wealth of recent work has focused on relieving the burden of network management by logically centralizing management tasks. 4D (13), 8 (13), NOX (10), Ethane (7, 6), Network Exception Handlers (12) and CONMan (3) all propose provide some centralized interface to configure a distributed and potentially diverse set of network elements. We continue with the idea of logically centralized management, and we add the novel contribution of using end hosts to implement network management.

Other work has attempted to simplify the networks themselves. OpenFlow (14) looks to add a small degree of programmability to existing switches enabling richer network policies without placing undue burden on the switches themselves. SEATTLE (13) instead aims to remove the need for routers by allowing Ethernet-switching to scale to whole enterprise networks.

Middleboxes have always been a contentious topic, but recent work has looked at how to embrace middleboxes and treat them as first-class citizens. In TRIAD (9) middleboxes are first-order constructs in providing a content-addressable network architecture. The Delegation-Oriented Architecture (16) allows hosts to explicitly invoke middleboxes, while NUTSS (11) proposes a novel connection establishment mechanism which includes negotiation of which middleboxes should be involved.

### 5 Conclusion

In today's world we want networks that just work. With the help of expensive middleboxes and large, well-trained IT departments, many large enterprise networks come close, but for small networks the story is less positive. For instance, our home networks do not protect web traffic from being delayed in long queues of file-sharing traffic, NATs interfere with peer-to-peer applications and as if that wasn't enough, even simple changes, like adding a second wireless access point to a house involve complex reconfiguration of many different devices. This is far from ideal.

Rather than require a vast array of expensive middleboxes and the associated management overhead, neither of which most small networks can likely afford, we have proposed a different tack. We presented a network architecture which leverages existing resources—namely, end hosts—to provide a network that just works based on shifting management toward the edge.

While we have designed this approach with small networks in mind, we believe that many aspects of the approach may be applicable in larger networks as well. Moving forward, we hope to put this approach into practice in home networks as well as extend the work to handle more diverse networks and management tasks.

### References

- [1] OpenWrt. <http://openwrt.org/>
- [2] ANDERSON, R. "Trusted Computing" Frequently Asked Questions. <http://www.cl.cam.ac.uk/~rja14/tpca-faq.html> August 2003.
- [3] BALLANI, H., AND FRANCIS, P. CONMan: A step towards network manageability. In *SIGCOMM* (2007).
- [4] CAESAR, M., CALDWELL, D., FEAMSTER, N., REXFORD, J., SHAIKH, A., AND VAN DER MERWE, J. Design and implementation of a routing control platform. In *NSDI* (2005).
- [5] CAI, Z., COX, A. L., AND NG, T. S. E. Maestros: A new architecture for realizing and managing network controls. In *LISA Workshop on Network Configuration* (2007).
- [6] CASADO, M., FREEDMAN, M. J., PETTIT, J., LUO, J., MCKEOWN, N., AND SHENKER, S. Ethane: Taking control of the enterprise. In *SIGCOMM* (2007).
- [7] CASADO, M., GARFINKEL, T., AKELLA, A., FREEDMAN, M. J., BONEH, D., MCKEOWN, N., AND SHENKER, S. SANE: A protection architecture for enterprise networks. In *USENIX Security* (2006).
- [8] GREENBERG, A., HALIMTYSSON, G., MALTZ, D. A., MYERS, A., REXFORD, J., XIE, G., YAN, H., ZHAN, J., AND ZHANG, H. A clean slate 4D approach to network control and management. In *CCR* (2005).
- [9] GRITTER, M., AND CHERITON, D. R. An architecture for content routing support in the internet. In *USITS* (2001).
- [10] GUDE, N., KOPONEN, T., PETTIT, J., PFAFF, B., CASADO, M., MCKEOWN, N., AND SHENKER, S. NOX: Towards an operating system for networks. In *CCR* (2008).
- [11] GUHA, S., AND FRANCIS, P. An end-middle-end approach to connection establishment. In *SIGCOMM* (2007).
- [12] KARAGIANNIS, T., MORTIER, R., AND ROWSTRON, A. Network exception handlers: Host-network control in enterprise networks. In *SIGCOMM* (2008).
- [13] KIM, C., CAESAR, M., AND REXFORD, J. Floodless in SEATTLE: A scalable ethernet architecture for large enterprises. In *SIGCOMM* (2008).
- [14] MCKEOWN, N., ANDERSON, T., BALAKRISHNAN, H., PARULKAR, G., PETERSON, L., REXFORD, J., SHENKER, S., AND TURNER, J. OpenFlow: Enabling innovation in campus networks. <http://www.openflowswitch.org/documents/openflow-wg-latest.pdf>, March 2008.
- [15] REXFORD, J., GREENBERG, A., HALIMTYSSON, G., MALTZ, D. A., MYERS, A., XIE, G., ZHAN, J., AND ZHANG, H. Network-wide decision making: Toward a waffer-thin control plane. In *HotNets* (2004).
- [16] WALSH, M., STRIBLING, J., KROHN, M., BALAKRISHNAN, H., MORRIS, R., AND SHENKER, S. Middleboxes no longer considered harmful. In *OSDI* (2004).
- [17] WATSON, G., MCKEOWN, N., AND CASADO, M. NetFPGA: a tool for network research and education. In *WARPP* (2006).
- [18] YAN, H., MALTZ, D. A., NG, T. S. E., GOGINEMI, H., ZHANG, H., AND CAI, Z. Tessera: A 4D network control plane. In *NSDI* (2007).



## Kurzfassung (Abstract)

- Legt das Thema der Arbeit in wenigen Sätzen ohne ins Detail zu gehen präzise dar und fasst die wichtigsten Punkte der Arbeit zusammen
  - **Kurzfassung != Inhaltsangabe**
  
- Entscheidungsgrundlage für Leser, die Arbeit zu lesen oder nicht
  - Die Kurzfassung ist also **wichtig** und sollte nicht 3 Minuten vor Abgabe geschrieben werden

Motivation	Autos sind heutzutage das Standard Fortbewegungsmittel.
Zu lösendes Problem	Durch die zunehmende Anzahl von Autos steigt die Umweltbelastung.
Lösungsansatz	Elektromotoren wurden als Alternative zu Verbrennungsmotoren vorgeschlagen.
Ergebnisse	Es gibt noch viele ungelöste Probleme die die Anwendbarkeit beeinträchtigen.
Fazit	Hybrid Motoren sind momentan die am besten geeignete Alternative zu Verbrennungsmotoren.



# Einführung (Introduction)

## □ Einführung

- Erklärt, warum das Problem wichtig ist und wie es zu anderen Problemen in Beziehung steht („Motivation“)
- Gibt eine präzisere Einleitung als der Abstract
- Berührt alle für das Verständnis wesentliche Punkte und Hintergrundinformationen („Grundlagen“)
- ➔ Erzeugt Interesse beim Leser
- Letzter Teil der Einführung: Überblick über den Aufbau des restlichen Papers
  - „Im folgenden Abschnitt 2 wird... Abschnitt 3 definiert die verwendeten Modelle...“
  - „The remainder of this paper is organized as follows: ...“

➔ Nach Titel und Abstract haben die Leser einen Überblick und wissen was weiter auf sie zukommen wird.



# Rumpf

- Kein einheitlicher Aufbau, variiert stark mit behandeltem Thema
- Beschränkung auf Inhalte die für das Thema wichtig sind
  - alles andere wird weggelassen oder referenziert
- Auf logischen und stringenten Aufbau achten!
- Auf eine ausgewogene Tiefe der einzelnen Teile achten
- Platziert das Thema in einem größeren konzeptionellen Bezugsrahmen



## Verwandte Arbeiten (Related Work)

- Gibt einen Überblick über andere, thematisch verwandte Arbeiten
  - **Essentielle Ideen** der anderen Arbeiten sollen vermittelt werden
  
- Die verwandten Arbeiten sollen dem vorgestellten Thema gegenüber gestellt werden
  - Vor- und Nachteile
  - Gemeinsamkeiten, ...
  - Erheblich wichtiger als ein bloßes Aufzählen „what others have done“
  
- Oft als letzter Teil vor Zusammenfassung und Ausblick



## Zusammenfassung (Conclusion)

- Fasst die Resultate zusammen
  - Aber **NUR** die Resultate, **NICHT** die ganze Arbeit noch mal
  - Dies ist **NICHT** noch mal eine Wiederholung des Abstracts!
- Diskutiert Ergebnisse aus einer größeren Perspektive, stellt einen größeren Zusammenhang her
- Kann ggf. Empfehlungen/Bewertung aussprechen
  - „Die vorgeschlagene Methode verspricht Erfolg.“
- Legt dar, was man vorher noch nicht wusste und aus der Arbeit gelernt hat
- Einschränkungen der Arbeit diskutieren
- **Neben Abstract der wichtigste Abschnitt einer Arbeit!**



# Übersicht

- Ziele
  - eines (Pro)Seminars
  - einer Ausarbeitung
  
- Struktur der Ausarbeitung
  
- **Tipps und Richtlinien zur Ausarbeitung**
  
- Tipps zum Vortrag



- Ausarbeitungen sind in deutscher oder englischer Sprache zu verfassen.
- Stichpunktartige Aufzählungen sind sparsam zu verwenden.
- Stets ganze Sätze verwenden.
  - Kein unnötig langer und verschachtelter Satzbau
  - Besser kurz und prägnant
- Keine umgangssprachliche Formulierungen
  - „Quasi“, „Oder so“, „Irgendwie so“, ...
- Die Verwendung der Ich/Wir-Form ist schlechter Stil
  - Falsch: In dieser Ausarbeitung gebe ich eine Einführung in Thema X ...
  - Richtig: Die vorliegende Ausarbeitung führt in das Thema X ein...





# Literaturrecherche

- Online-Quellen
  - Google Scholar: <http://scholar.google.com/>
  - Citeseer: <http://citeseer.nj.nec.com/>
  - IEEEXplore: <http://ieeexplore.ieee.org>
  
- interessante Verlage / Herausgeber:
  - IEEE / ACM (auch Online-Datenbank)
  - Kluwer, Baltzer, Addison-Wesley, Springer, Artech House, CRC Press
  
- Gute Quellen (glaubwürdig):
  - Alles was „begutachtet/reviewed“ wurde, wie Bücher, Journal papers, Conference papers
  - In Spezialfällen „Technical Reports“ direkt von den Herstellern
  
- Schlechte Quellen:
  - Heise-Newsticker
  - Wikipedia
  - i.A. Webseite XYZ



## Zugriff auf Literatur

- Über Webseite der Autoren
  - Autoren stellen zum Teil Paper auf eigene Homepage
  
- Über Springer, ACM, IEEE
  - Download meist kostenpflichtig
  - TUM verfügt über vollen Zugriff
  - Verwendung eines Proxy-Servers notwendig:  
**[wwwcache.informatik.tu-muenchen.de](http://wwwcache.informatik.tu-muenchen.de), Port 8080**
  - Zugriff auf diesen Proxy ist auf das TUM-Netz beschränkt
  - Alternative: **<http://pac.lrz-muenchen.de/>**



## Verweise / Referenzen

- Wird im Text auf einen anderen Teil Bezug genommen, geschieht dies mit genauer Kapitelangabe:
  - Falsch: Wie weiter oben beschrieben ...
  - Richtig: Wie in Abschnitt 2.1 beschrieben ...
  
- Auf Grafiken innerhalb der Ausarbeitung
  - Alle Grafiken besitzen eine Nummer und kurze Beschreibung
    - Abbildung 1: Architekturüberblick (Figure 1: Architectural Overview)
  - Jede Abbildung muss an passender Stelle referenziert und erklärt werden
    - Falsch: Die oben abgebildete Architektur unseres ... besteht aus ...
    - Richtig: Die Architektur des Systems für ... , siehe Abbildung 5, besteht aus ...
  
- Auf Texte / Bücher / Paper anderer Autoren
  - Falsch: Waitzmann beschreibt in seinem im April 1990 erschienenen RFC „Standard for the transmission of IP datagrams on avia” ...
  - Richtig: Waitzmann [6] beschreibt ...



## Was beinhalten Quellenangaben (References)?

- Vollständige Quellenangabe!
  - Autor(en)
  - Titel
  - Als was erschienen? (Buch, Zeitschriftenartikel, Konferenzbeitrag, Teil eines Buches, technischer Bericht)
    - Titel der Zeitschrift, des gesamten Buches, etc.
    - Ggf. Herausgeber
    - Seitenzahlen
  - Erscheinungsjahr, -monat und –ort
  - Verlag
  - URLs können hilfreich sein, sollten aber möglichst stabil sein
    - Schwierig vorauszusehen
  - Zusätzliche Informationen nach Bedarf



## Wie sehen Quellenangaben (References) aus?

### ❑ Falsch:

[6] RFC1149 - Standard for the transmission of IP datagrams on avia  
<http://www.faqs.org/rfcs/rfc1149.html>

### ❑ Richtig:

[6] **D. Waitzman**, „Standard for the transmission of IP datagrams on avia”,  
**RFC 1149, IETF, April 1990**

### ❑ Als BiBTeX:

```
@MISC{Waitzman1990,  
author = {D. Waitzman},  
title = {{Standard for the transmission of IP datagrams on avia}},  
howpublished = {RFC 1149, IETF},  
month = {April},  
year = {1990},  
}
```

### ❑ Hilfreiche Tools zur Generierung von Bibtex:

- Online Bibtex Generator: <http://www.odyle.net/mma/tools/tex/bibtex/bibgen.php>
- Jabref: <http://jabref.sourceforge.net/screenshots.php>



- Zitate sind sparsam zu verwenden ...
- ... und haben oft den Charakter einer Definition
  
- Zitat hat immer eine Referenz auf eine Textquelle (Literaturangabe)
- Literaturangaben sollen Auffinden der Quellen möglichst einfach machen
  
- Literaturangabe wird wie üblich eingebunden:
  - Bsp: Waitzman [6] definiert einen Vogel als „vollständig biologisch abbaubaren, autonom agierenden Flugkörper mit Segelflugeigenschaften“.



## Spezielles für den Lehrstuhl I8

- Verwendung der Templates auf der Seminarhomepage sind Pflicht!
  - Für Ausarbeitung, Vortrag und Reviews
  - In Latex ist das Style-Sheet für BiBTeX bereits vorgegeben
  - BiBTeX-Addon für Word: <http://www.bibtex.org/Using/de/>
  
- Zum Vortrag bitte zusätzlich ein PDF-File mitbringen
  
- Alle Ausarbeitungen werden automatisch auf Plagiarismus geprüft!



# Übersicht

- Ziele
  - eines (Pro)Seminars
  - einer Ausarbeitung
  
- Struktur der Ausarbeitung
  
- Tipps und Richtlinien zur Ausarbeitung
  
- **Tipps zum Vortrag**





## Tipps zum Vortrag

- An die Zuhörer denken!
  - Was würde euch interessieren/langweilen, wenn ihr die Zuhörer eures Vortrags wärt?
  
- Speziell im Hacking-Seminar:
  - Keine ewig langen Listen von irgendwas (z.B. Sicherheitslücken) durchgehen!
  - Auf wenige Punkte konzentrieren, diese aber im Detail erklären.
    - Mit Beispielen und Hintergründen, so dass die Kernthemen für die Zuhörer wirklich nachvollziehbar sind.
    - ... und dann vielleicht noch ganz kurz eine Folie „was es sonst noch so gibt“.
  
- Orientierung der Zuhörer
  - Inhaltsübersichtsfolien sind etwas Gutes!
  - ABER: Viele Vortragende tendieren dazu, dort schon den halben Vortrag zu erzählen.
    - „In der Einleitung werde ich ins Thema einführen, ...“
  
- Die Zeit einhalten!
  - Am besten den Vortrag zu Hause trocken üben.



## Weitere Informationen

- Bücher zum Thema Schreiben, Organisation von Texten, Stil
  - Joseph M. Williams, *Style: Ten Lessons in Clarity and Grace*, ISBN: 0321024087, und andere Bücher von diesem Autor
  - Thomas S. Kane, *The New Oxford Guide to Writing*, ISBN: 0195090594
  - M. Alley, *The Craft of Scientific Writing*, ISBN: 0387947663
  - Wolf Schneider, *Deutsch für Kenner. Die neue Stilkunde*, ISBN: 3492222161
  
- Web-Seiten
  - Überblick: <http://www.cs.emu.edu/afs/cs.cmu.edu/user/mleone/web/how-to.html> mit Links zu einer Reihe von Artikeln.

