**Einleitung**

1. Internet impasse
2. Einführung von Virtualisierung (löst den impasse)
   1. Warum ist Virt. So toll?
   2. VNE-Problem
3. Bislang nur unter Performance-Aspekten betrachtet
4. Sicherheitsrisiken bestehen
5. Begriffe
6. Gliederung

**Internet impasse**

Ein Konzept dem Internet Impasse [PAPER: Internet Impasse, 2005] mit flexibler Architektur [WAS IST DAS?] [und Handhabbarkeit] zu begegnen, wurde in der Netzwerkvirtualisierung gefunden.

Großen Hostinganbietern / Serviceprovidern, die ihr Netzwerk häufig umkonfigurieren, wird …

**Netzwerkvirtualisierung**

* Vorteile von Netzvirtualisierung
  + Abstraktion der physischen Hardware -> Beinahe unabhängig davon
  + Maschinen erstellen, verändern, migrieren, zurücksetzen genauso leicht wie der Umgang mit/das Kopieren von Dateien. -> Flexibilität, dynamischere Nutzung
  + Sandboxing
    - Man kann sich „nichts kaputt machen“
    - Testumgebungen. Neuen Code in verschiedenen Umgebungen (Win, Linux, ohne SW-Developer-Kit etc) testen ohne zusätzliche Hardware.
    - Deployment neuer Architekturen erleichtert. -> Overcoming the internet impasse.[Angabe]
  + Software und Hardware als on-demand Ressourcen
  + Hostseitig:
    - Geringerer Wartungsaufwand und IT-Kosten
    - Skalierbarkeit (nicht streng durch Hardware limitiert)
    - Automatisierbar: Kunde kann sich ohne menschliche Interaktion weitere Ressourcen bei Bedarf hinzubuchen
  + Kundensicht:
    - Unternehmen bezahlen nur für die benötigten Ressourcen, wenn sie gerade gebraucht werden. Hochqualitative Hardware zu einem Bruchteil ihres Preises. Keine ungenutzte Hardware. Einfaches Vergrößern der IT-Landschaft. (Weitere Server auf Knopfdruck hinzu- oder abbuchen)
    - Einfaches Skalieren (z.B. in Zeiten hoher Last)
  + Service orientierte Hardware
  + Mehr Nutzer pro phys. Host; Koexistenz mehrerer heterogener virt. Netzwerke auf selben Substratnetzwerk
  + Befördert die Flexibilität der Netzwerkarchitektur
  + -> Automatisierbar (programmierbare Netzwerkumgebung)
  + Begegnet der Behinderung in der Entwicklung innovativer Netzwerkarchitekturen[PAPER: Overcoming the impasse]
    - Internet Ossification Problem.
    - Z.B. durch anbieten verschiedener Protokollfamilien
  + Unterteilung in Service-Provider (SP) und Infrastructure-Provider (InP) möglich. Freiheiten durch jeweils unabhängige Technologieentscheidungen [wang2016towards]
* Was ist Netzwerkvirtualisierung?
  + Gruppen zusammengehöriger Virt. Maschinen laufen auf verschiedenen, ggfs. räumlich weit getrennten physischen Maschinen genauso als ob sie ihr eigenes (virtuelles) Netzwerk hätten.
  + Netzwerkvirtualisierung basiert auf Knoten- (z.B. Xen)[] und Linkvirt.
* Umsetzungsmöglichkeiten
  + IP-Ebene? -> Problem mit nicht-IP-Protokollen und IP-Support-Protokollen (ARP) -> Verkapselung nur mit Performanceeinbußen.
* Ziele:
  + Automatisierung
  + Software einfacher nutzbar Machen und Geschäftsprozesse befördern
  + Outsourcing von Computing, Storage, content und Network   
    Diese Übertragung der Verantwortung führte zu einer Anpassung des Risikomanagements.
  + => C.I.A. -> Security wird wichtig.
    - Bekannte Sicherheitsmechanismen wie Verschlüsselung, Firewalls, Intrusion Detection Systeme etc. können auf den virtuellen Komponenten des Netzwerks implementiert werden. Die Sicherheit von Nutzerdaten könne dadurch aber wegen der heterogenen und stark dynamischen Struktur virtueller Umgebugen jedoch nicht garantiert werden und Vorteile der Netzwerkvirtualisierung dürften durch den zusätzlichen Overhead verloren gehen. [PAPER: Virt. Netzwork Embedding throug Sec. Risk. Aware a. Opt.]
    - -> Eine mögliche Lösung hierzu ist das Integrieren von Sicherheitsaspekten bereits in die Zuordnung von virtuellen zu physischen Knoten und Links. Dieser VN Embedding Prozess stellt eine der größten Herausforderungen in der Netzwerkvirtualisierung dar.[PAPER: Fischer 2013]
    - Entsprechend ihrer Sicherheitsanforderungen sollten virtuelle Netzwerke auf geeignete Substratknoten entsprechender Firewall etc. gemappt werden.  
      Werden virtuelle Netze entsprechend ihrer Sicherheitsanforderungen bereits auf Substratknoten mit hinreichender Schutzfunktion wie beispielsweise Firewall gemappt, so kann Overhead durch zusätzliche Sicherheitstechnik reduziert werden.
    - Derzeit wird versucht [ANGABEN] bereits im VNE-Algo. diese Probleme anzugehen (SVNE Security-aware VNE|security by design). Nicht alle Probleme/Sicherheitsrisiken können aber auf diese Weise adressiert werden. (= Überleitung zur Gliederungs und zweiten Kapitel?)
* Logische Abstraktion der tatsächlichen Hardware
* Zwei Schritte: Logische Repräsentation der gewünschten Konfiguration und Mapping auf physische Ressourcen (VNE-Problem)
  + VNE: Virtual Network Embedding
    - Netzwerkvirtualisierung -> Knotenvirtualisierung und Linkvirtualisierung  
      => VNE: Zuteilung von virt. zu phys. Hosts (i.d.R. Erzeugung von Virtuelle Maschinen) und von virt. zu phys. Links/Verknüpfungen/Verbindungen/Pfade
    - Design der VM Interkonnektivität.
  + *-> SIEHE KAPITEL VON GERHARD*
* VNE-Problem ist NP-hart. [BELEG/ZITAT?]

**Begriffe**

* Alles, was für die restliche Arbeit benötigt wird!
* VM-Monitor / Hypervisor
* (S)VNE
* Gast- und Hostsystem
* Substratnetzwerk vs. Virt. Netzwerk
* Phys. Host = Substratnetzknoten, Virtuelle Maschine = virt. (Netz-)Knoten
* Netzwerkarchitektur
* Virt. Netzwerk = Sammlung von virt. Knoten und virt. Links
* Ressourcen: Bandbreite, CPU, Haupt- und Hintergrundspeicher
* Side channel attack
* Verwendung von ‚Risiko‘ und ‚Verwundbarkeit‘
* Abgrenzung zum Cloudcomputing?
* Was verstehen wir in dieser Arbeit unter ‚Sicherheit‘?

**Ausformulierte Gliederung**