

Mobile Transportschicht

Seminar: Mobile Computing

WS 2004/2005

Gunnar Adler

Gliederung

- **Einleitung / Traditionelles TCP**
 - TCP Mechanismen
- TCP Optimierungen
 - Split Connection Verfahren
 - Fast Retransmit / Fast Recovery
 - Einfrieren der Verbindungsparameter
 - Selektive Übertragungswiederholung
 - Transaktionsorientiertes TCP
 - Header Compression

Einleitung: Transportschicht

- Unterstützung von Mobilität in höheren Schichten Transportschicht notwendig
- Transportschicht stellt Anwendungen Kommunikationskanal zur Verfügung
- Anwendungen werden über Ports adressiert
- Protokolle TCP & UDP setzen auf Vermittlungsprotokoll IP auf

Einleitung: TCP & UDP

- UDP bietet im Gegensatz zu TCP keinerlei Garantien über die Auslieferung der Daten an die Anwendung
- Bei Paketverlust nimmt TCP eine Stausituation an und drosselt die Senderate, UDP zieht sich nicht zurück
- TCP Mechanismen bestimmen Leistungsfähigkeit im mobilen Umfeld

Klassisches TCP

- Für Festnetze mit festen Endsystemen entworfen → kaum Übertragungsfehler
- Grund für Paketverlust ist fast immer Stau an einem Netzknoten
- Senderate wird stark abgesenkt um den Stau schnell abzubauen
- TCP garantiert Aufteilung der Bandbreite unter allen Nutzern

Slow Start

- Verhalten nach Stauererkennung
 - Sender berechnet fortlaufend Staufenster
 - Staufenster wird bei Eintreffen von Bestätigungen exponentiell vergrößert
 - Übergang zu linearem Wachstum an Stauschwellenwert (Congestion Threshold)
 - bei ausgebliebenen oder mehrfachen Bestätigungen wird Stauschwellenwert auf Hälfte des aktuellen Staufensters und Slow Start Mechanismus wiederholt sich

Fast Retransmit / Fast Recovery

- Empfang von Bestätigungen deutet auf kurzfristigen Stau oder Übertragungsfehler
- Sender überträgt fehlende Pakete sofort erneut (Fast Retransmit) und fährt mit hoher Senderate fort (Fast Recovery)
- Staufenster muss nicht verkleinert werden

Auswirkungen auf Mobilität

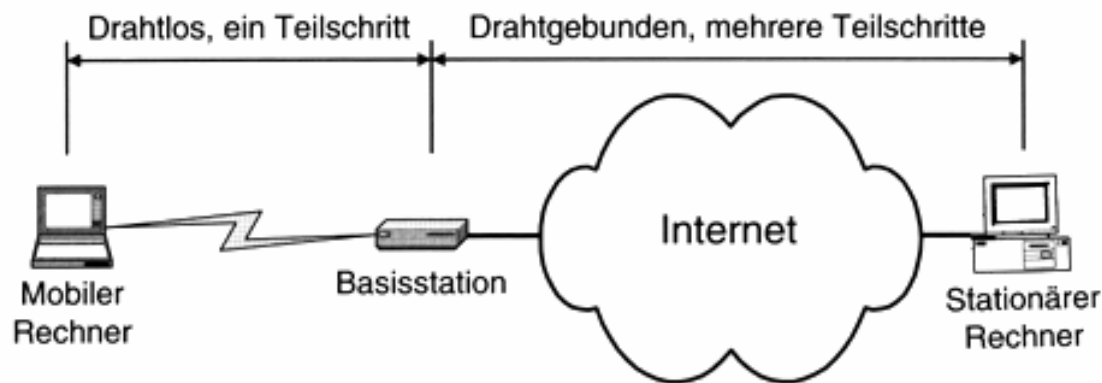
- Slow Start kann Leistungsfähigkeit von mobilen Geräten stark verschlechtern
 - häufiger Paketverlust aber kein Stau durch:
 - Vergleichsweise hohe Bitfehlerraten
 - Übergang zwischen Anschlusspunkten
 - klassisches TCP kann nicht zwischen Stau und Verlusten durch Mobilität unterscheiden
 - Erweiterungen müssen kompatibel sein und dürfen Stauverhalten nicht gefährden

Gliederung

- Einleitung / Traditionelles TCP
 - TCP Mechanismen
- **TCP Optimierungen**
 - Split Connection Verfahren
 - Fast Retransmit / Fast Recovery
 - Einfrieren der Verbindungsparameter
 - Selektive Übertragungswiederholung
 - Transaktionsorientiertes TCP
 - Header Compression

Split Connection Verfahren

- Teilung der Verbindung in Festnetzteil und drahtlose Strecke
 - klassisches TCP zwischen stationärem Rechner und Basisstation und spezielles TCP zwischen Basisstation und mobilem Rechner

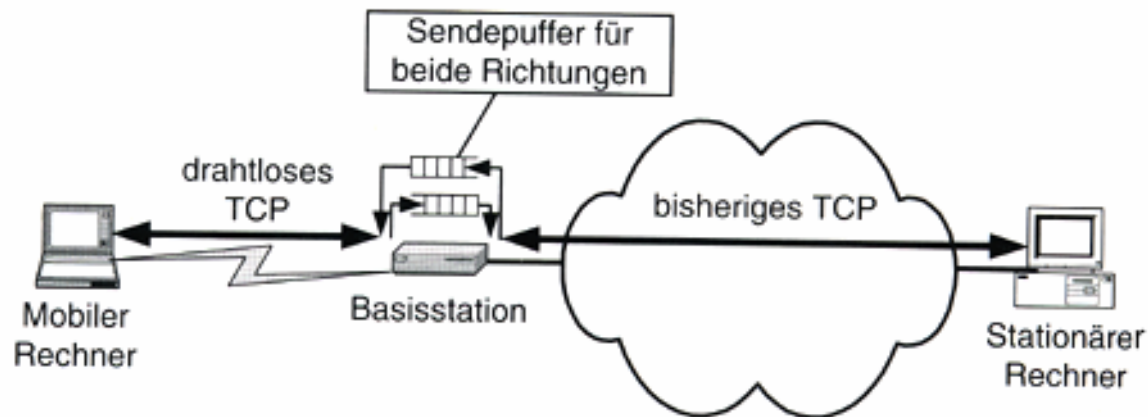


Indirektes TCP (I-TCP)

- Basisstation/Fremdagent handelt als Stellvertreter (Proxy) und leitet Daten in beide Richtungen weiter
 - Daten werden zwischengespeichert
 - Bestätigung an Sender erfolgt unmittelbar
 - erst dann werden Daten zum Empfänger weitergeleitet

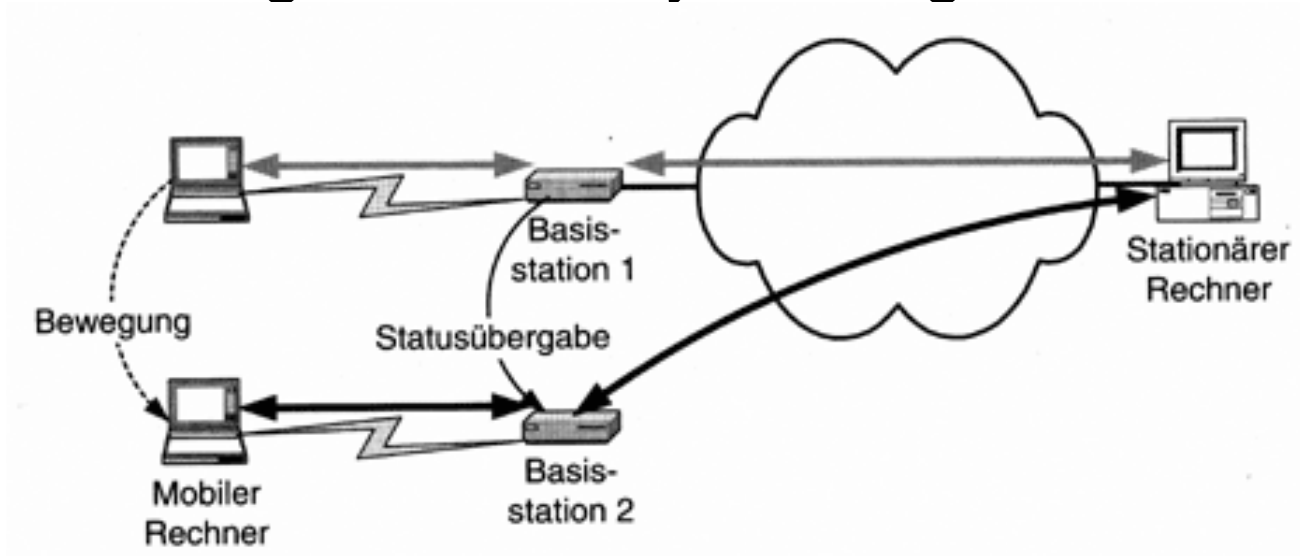
Indirektes TCP (I-TCP)

- Übertragungsfehler auf drahtloser Strecke werden vom Stellvertreter behandelt
→ lokale Übertragungswiederholung



Indirektes TCP (I-TCP)

- Wechsel des Zugangspunktes
 - Mobile IP regelt Umleitung der Datenpakete
 - Bereits im Proxy gepufferte Daten werden an neuen Agenten/Proxy weitergeleitet



Indirektes TCP (I-TCP)

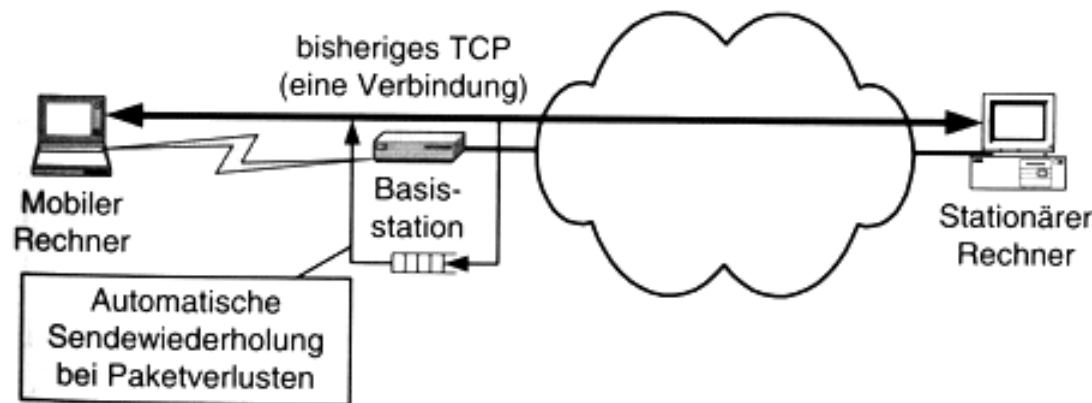
- Vorteile:
 - keine Änderung an TCP im Festnetz
 - Änderung im Funknetz nicht zwingend
 - Übertragungsfehler pflanzen sich nicht fort
 - schnellere Durchführung von Übertragungswiederholungen
 - Einsatz von anderen Transportschicht Protokollen im drahtlosen Teil möglich

Indirektes TCP (I-TCP)

- Nachteile:
 - Verlust der Ende-zu-Ende Semantik von TCP
 - vergrößerte Verzögerung bei Wechsel des Zugangspunktes
 - Stellvertreter muß vertrauenswürdig sein (volle Integration bei Verschlüsselung)
 - bei Wechsel des Zugangspunktes muss neuer Agent auch I-TCP unterstützen

Snooping TCP

- Transparente Erweiterung
 - Pakete in Richtung des mobilen Rechners werden zwischengespeichert
 - Paketfluss wird in beide Richtungen abgehört
 - Agent sendet keine Empfangsbestätigungen



Snooping TCP

- Vorteile:
 - Ende-zu-Ende Semantik bleibt erhalten
 - keine Änderung an TCP notwendig
 - keine Zustandsübergabe bei Wechsel des Zugangspunktes
 - bei Wechsel des Zugangspunktes muss neuer Agent diese Erweiterung nicht zwingend unterstützen

Snooping TCP

- Nachteile:
 - geringe Isolation der drahtlosen Verbindung
 - Einsatz negativer Bestätigungen (NACK) erfordert zusätzliche Mechanismen
 - Mitlesen der Pakete der schlägt beim Einsatz von Ende-zu-Ende Verschlüsselungen fehl
 - (lokales) Wiederholen von Daten bei vielen Sicherheitsmechanismen nicht möglich

Mobile TCP (M-TCP)

- Optimierung des TCP Verhaltens bei längeren Verbindungsunterbrechungen
 - Segmentierte Verbindung ähnlich wie I-TCP
 - M-TCP setzt niedrige Fehlerraten voraus
 - Keine Zwischenspeicherung der Daten
 - Überwachungsknoten (Supervisory Host) beobachtet alle Pakete zum und alle Bestätigungen vom mobilen Rechner

Mobile TCP (M-TCP)

- Funktionsweise:
 - bleiben Bestätigungen vom mobilen Rechner aus, wird davon ausgegangen dass gerade keine Verbindung zum Netz besteht
 - Überwachungsknoten drosselt stationären Rechner, indem Sendefenster auf null gesetzt wird (Persistenter Modus)
 - wenn die Verbindung wieder besteht wird Sendefenster mit altem Wert wieder eröffnet

Mobile TCP (M-TCP)

- Vorteile:
 - erhält Ende-zu-Ende Semantik von TCP
 - unnötige Übertragungswiederholungen werden vermieden wenn mobiler Rechner getrennt ist
 - keine Zustandsübergabe bei Wechsel des Zugangspunktes
 - Keine Änderungen an TCP Implementation des stationären Rechners notwendig

Mobile TCP (M-TCP)

- Nachteile:
 - durch fehlende Zwischenspeicherung sind Paketverluste auf der drahtlosen Verbindung für den stationären Rechner sichtbar
 - modifiziertes TCP auf mobiler Seite muß auf allen beteiligten Geräten installiert sein
 - fehlendes Slow Start auf mobiler Seite macht Bandbreitenverwalter notwendig

Fast Retransmit / Fast Recovery

- Bei Wechsel zu neuem Fremdagenten
- Schnelle Übertragungswiederholung kann durch duplizierte Bestätigungen künstlich hervorgerufen werden
 - Kommunikationspartner wird sofort in Fast Retransmit gezwungen (kein Slow Start)
 - vorige Senderate wird fortgesetzt

Fast Retransmit / Fast Recovery

- Vorteile:
 - nur relativ kleine Änderung in der Software des mobilen Gerätes notwendig
 - weder Fremdagenten noch Kommunikationspartner müssen modifiziert werden

Fast Retransmit / Fast Recovery

- Nachteile:
 - ungenügende Isolation von Paketverlusten
 - bei länger andauernden Wechseln beginnt der Kommunikationspartner wahrscheinlich schon mit erneutem Senden der Daten
 - Paketverlust durch Übertragungsprobleme auf drahtloser Strecke wird nicht berücksichtigt

Einfrieren der Verbindungsparameter

- Umgang mit längeren Unterbrechungen
 - MAC Schicht informiert TCP über bevorstehende Unterbrechungen, damit TCP nicht von einem Stau ausgeht
 - TCP stellt Senden ein und behält aktuellen Zustand bei und benachrichtigt ggf. den Kommunikationspartner
 - bei wiederhergestellter Verbindung informiert MAC Schicht TCP → setzt Arbeit fort

Einfrieren der Verbindungsparameter

- Vorteile:
 - TCP Verbindungen können nach längerer Unterbrechung fortgesetzt werden
 - unabhängig von anderen TCP Mechanismen
→ verschlüsselte Pakete möglich
- Nachteile:
 - Software muss auf beiden Kommunikationspartnern angepasst werden

Selektive Übertragungswiederholung

- indirekt kann die Wiederholung einzelner Pakete angefordert werden
 - Möglichkeit zur Bestätigung einzelner Paket
 - fehlende Pakete werden erneut übertragen

Selektive Übertragungswiederholung

- Vorteile:
 - Sender überträgt nur tatsächlich verlorene Pakete erneut → spart Bandbreite
 - bereits von vielen TCP Implementierungen unterstützt
- (Nachteile):
 - Software auf Empfängerseite etwas komplexer

Transaktionsorientiertes TCP (T/TCP)

- Kombination von Datenpaketen mit Paketen für Verbindungsauf- und -abbau
- Vorteil:
 - Zahl der Pakete wird bei kurzen Anfragen drastisch reduziert
- Nachteil:
 - kein normales TCP → Änderungen bei allen Kommunikationspartnern notwendig

Header Compression

- Reduzierung von Mehraufwand durch große aber fast identische TCP und IP Header
- lediglich Änderungen werden versendet
- nützlich für verzögerungsempfindliche Anwendungen mit Vielzahl kleiner Pakete