Do current WWW Protocols work over Wireless and Small Screen Devices?

Gabriel Montenegro
gab@sun.com
Sun Microsystems Laboratories
IAB Wireless Internetworking Workshop
3/2000
yes

• questions?
Outline

- Some Background
- Network Characteristics
- Implementation Constraints
- Handling platform challenges
- Handling network challenges
- Going forward
Some Background

- Mobile Network Computer Reference Specification (MNCRS) - aka Mobile Dodo
  - Mobile Communications working group
  - Mobile IP (home addr discovery, chained/surrogate tunnels)
  - Messaging Middleware
  http://www.mncrs.org/
  http://computer.org/internet/v2n1/mncrs.htm

- Open networking protocols FROM the IETF
  - Performance Implications of Link Characteristics (PILC)

- Open Web Layers FROM W3C
  - Mobile Access Interest Group
  http://www.w3.org/Mobile/Group/IG/
Network Characteristics

- Long Thin *Wireless* Networks (GPRS, CDPD, TDMA, GSM, CDMA, Metricom, DoCoMo, PHS, ...)
- Latencies typically >400ms
- Low Bandwidth (<30Kbps), small delay*bandwidth product(increasing)
- Typical architecture:
  - mobile device, connected via a long thin wireless link to an...
  - intermediate system (base station/proxy)
  - legacy servers
- Stationary or semi-stationary usage
- **BUT:** 10/100BT/802.11 (and no proxy) is also possible!
Implementation Constraints

Devices are less than 512KB:

- Size of embedded stacks (usually TCP/IP <15KB)
  - KA9Q (Phil Karn) ~ 12KB
    - TCP: 10KB
    - IP: 1.8KB
    - PPP: 14KB
  - Smartcode Embedded NetCore IP ~ 14KB
    http://www.smartcodesoft.com/service/service.html
  - STN~ 30KB with PPP
    http://www.stnc.com
  - IPv6 functional implementation in ~10KB of C

- Amount of available bandwidth
  - 512bps, 4Kbps, 9.6Kbps, 14.4Kbps, 20Kbps
Handling platform challenges

- Stack Autotuning to set recv/xmit buffers
  http://www.psc.edu/networking/auto.html
- TCP control block interdependence (RFC 2140)
- Proxies: an **optional** optimization
  - Offload public-key operations? Risky and perhaps unnecessary
  - Offload image processing/filtering
  - Firewall traversal
  - Interface to back-end email/calendar/dbases
  - v4/v6 interface?
  - Amortize your tcp connection establishment
Handling platform challenges (2)

W3C Mobile Access Interest Group

- HTML 3.2 subsets

- Compact HTML

http://www.w3.org/TR/1998/NOTE-compactHTML-19980209/

- HTML 4.0 Guidelines for Mobile Access

http://www.w3.org/TR/NOTE-html40-mobile/

- Just use pdQBrowser, HandWeb, Palmscape...

- AvantGo model

- I-Mode !!! 4million in under a year, content explosion
Handling network challenges

In long-thin networks (RFC 2757), SLOW, ERROR and LINK, independent of V4 versus V6:

- Improve the link with FEC and retransmits
- Implement error-resilient header compression (RFC2507,RFC2509)
  - V6 is more compressible (no IP header checksum)
- Path MTU discovery (RFC1191)
- TCP’s initial window to 2 segments (RFC2414/2581)
- ACK without delay the first segment in a new connection (expensive?)
- for 3G: SACK (RFC2018) and DSACK (or NewReno)
- Infrastructure: ECN (RFC2481) and RED (RFC 2309)
Going Forward (1 of 3)

- HTTP1.1
  - NO CLEAR FUTURE OPTION HERE!!!!!
  - persistent connections (vs T/TCP)
  - “Content-Encoding: deflate” and "Accept-Encoding: deflate”
  - http://www.w3.org/Protocols/HTTP/Performance/Pipeline.html

- TLS allows for compression!

- Steve Bellovin’s TCP filters include compression

- TCP filters-->
  ”IPCP for TCP”?

- Generalized XML tokenization (like WML’s)
  - using any of the above compression negotiations

- Handle options better in header compression

- UDP-Lite for voice?
Going Forward (2 of 3)

- Modularization
  - XHTML Basic
  - XHTML Mobile Profile

- Transport options
  - T/TCP?
  - Will http1.1 deliver?
  - SCTP?
  - Endpoint Congestion Management
  - Ensemble TCP?
  - TCP for wireless mostly under control (PILC item), but the real problem is:

HTTP!!! HTTP-NG is dead, where to go now?
Going Forward (3 of 3)

- **Security for small devices**
  - TLS in small devices (definitely doable, prototyping work under way)
  - RSA may have some advantages over ECC after all: crypto.stanford.edu/~dabo/abstracts/PalmPilotWallet.html
  - NTRU?

- **Non-Congestion related Loss**
  - “network unreachable”
  - “freeze tcp” - advertising Rx window of 0 and growing it back up
  - “network reachable”
  - “corruption experienced”
Conclusion

Open protocols:
- are small enough
- are adaptable and most definitely *NOT* broken
- benefit from open review by world experts (IETF, W3C)
- are public and openly implementable
- allow optional proxy services, but do not require them
- do not break the end-to-end principle (so IP security would work, for example)
- are scalable much beyond current wide-area wireless bandwidths