Security Design

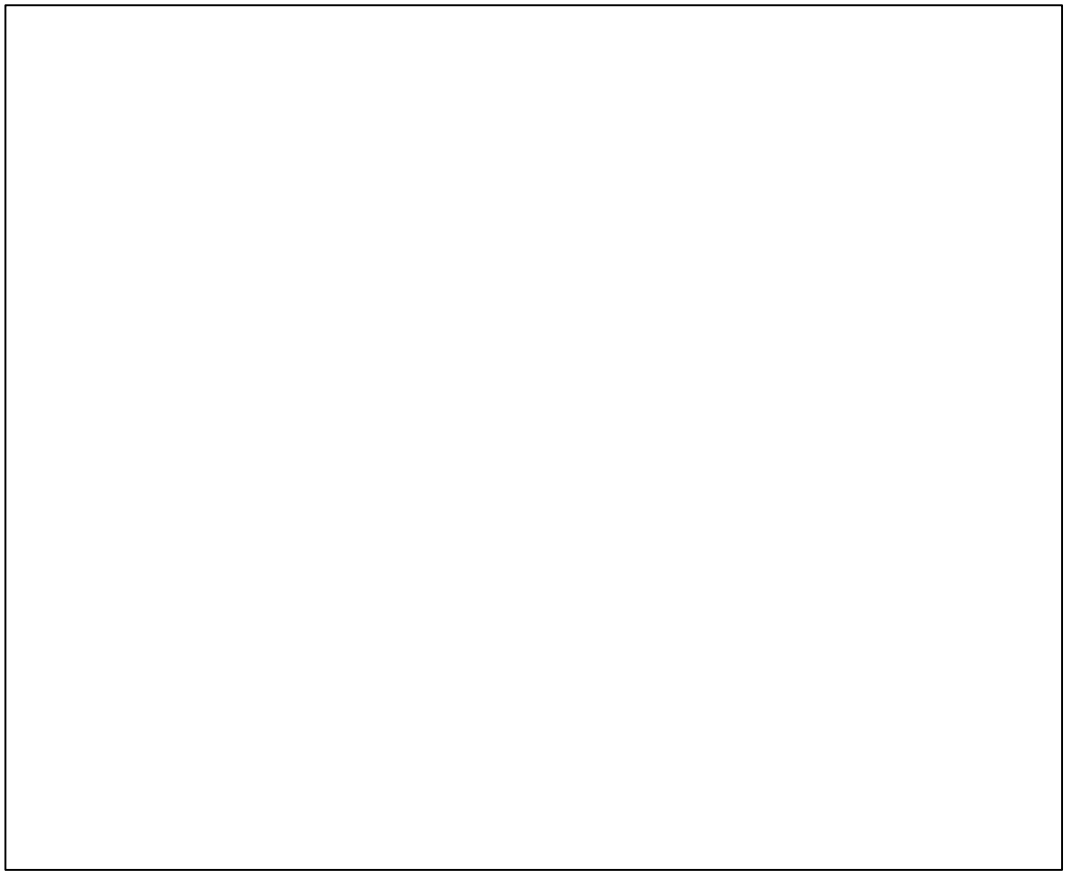
for
Application Protocols

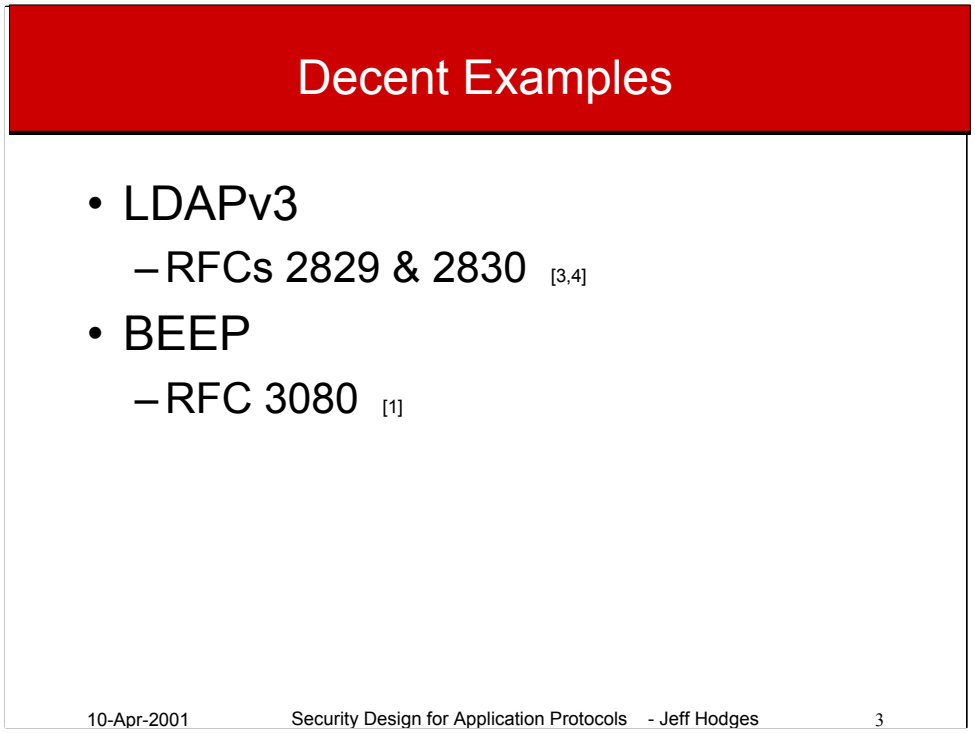
v0.2

Jeff Hodges
10-Apr-2001
jhodges@oblix.com
http://www.stanford.edu/people/Jeff.Hodges

Syllabus

- Decent Examples
- Protocol Review
- Threats
- Security Mechanisms as a Function of Anticipated Threats & stuff
- Example: LDAP Protocol Security Features
- LDAP Security Features - Illustrated
- What's Lacking in LDAP Security Features?
- Decent Examples, again



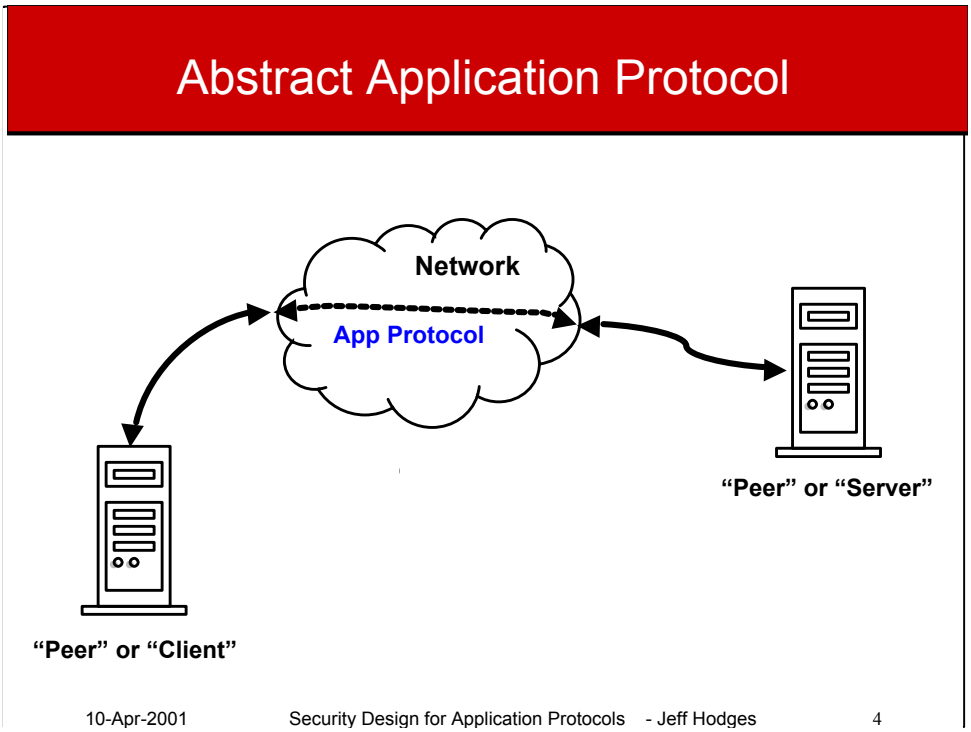


These two protocols cleanly rely upon incorporation of authentication mechanisms via SASL [8], and also incorporate a notion of establishing a TLS-based [9] secure session layer without using a separate, dedicated port.

Recent HTTP RFCs add similar capabilities to that protocol, although there is not an overall specification tying those recent capabilities to the original HTTP RFCs (2616, 2617).

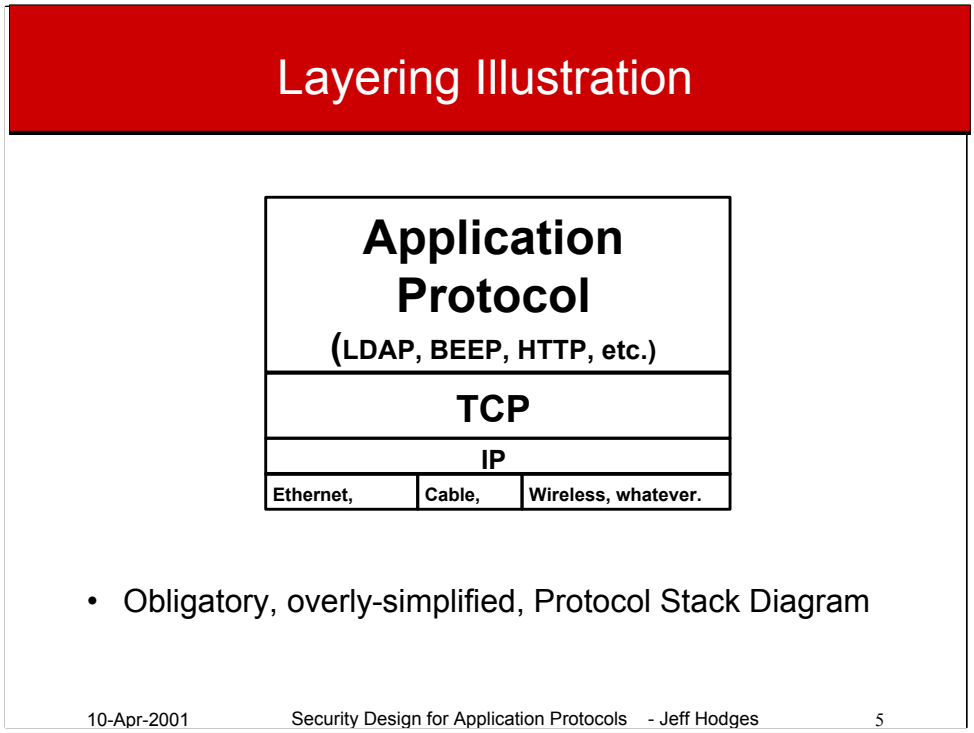
LDAPv3 has a similar specification issue as HTTP, but it will be addressed once an RFC based on http://www.ietf.org/internet-drafts/draft-ietf-ldapbis-ldapv3-ts-00.txt is issued [2].

Additionally, LDAPv3 isn't quite as "clean" as BEEP [1] in that it still has the notion of a protocol-specific, simple in-the-clear username & password authentication, also known as a "simple BIND", or a "BIND of the simple flavor" (as opposed to a SASL-based BIND); see [14].

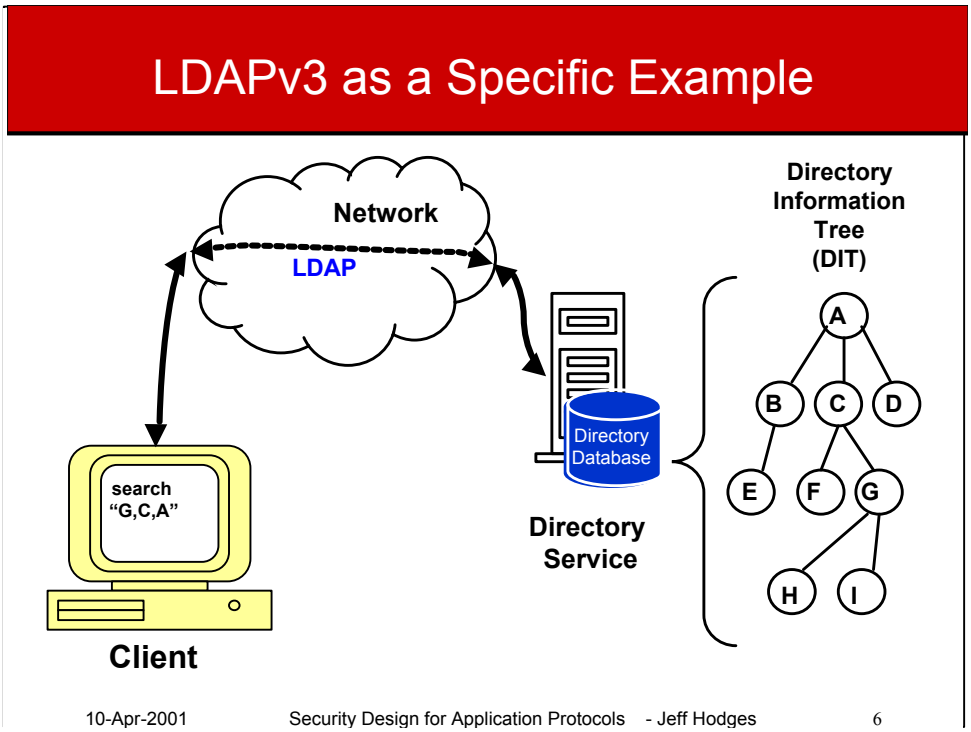


Ok, don't spend much time on this slide, it's pretty basic ;-)

The next slide is also a simple depiction of "what's going on under the hood" of this picture.



This is an illustration of "what's going on under the hood" of the previous picture -- a fair amount of stuff, all told – though much is still "hidden" in this picture, e.g. interactions with the DNS and possibly various intermediaries.



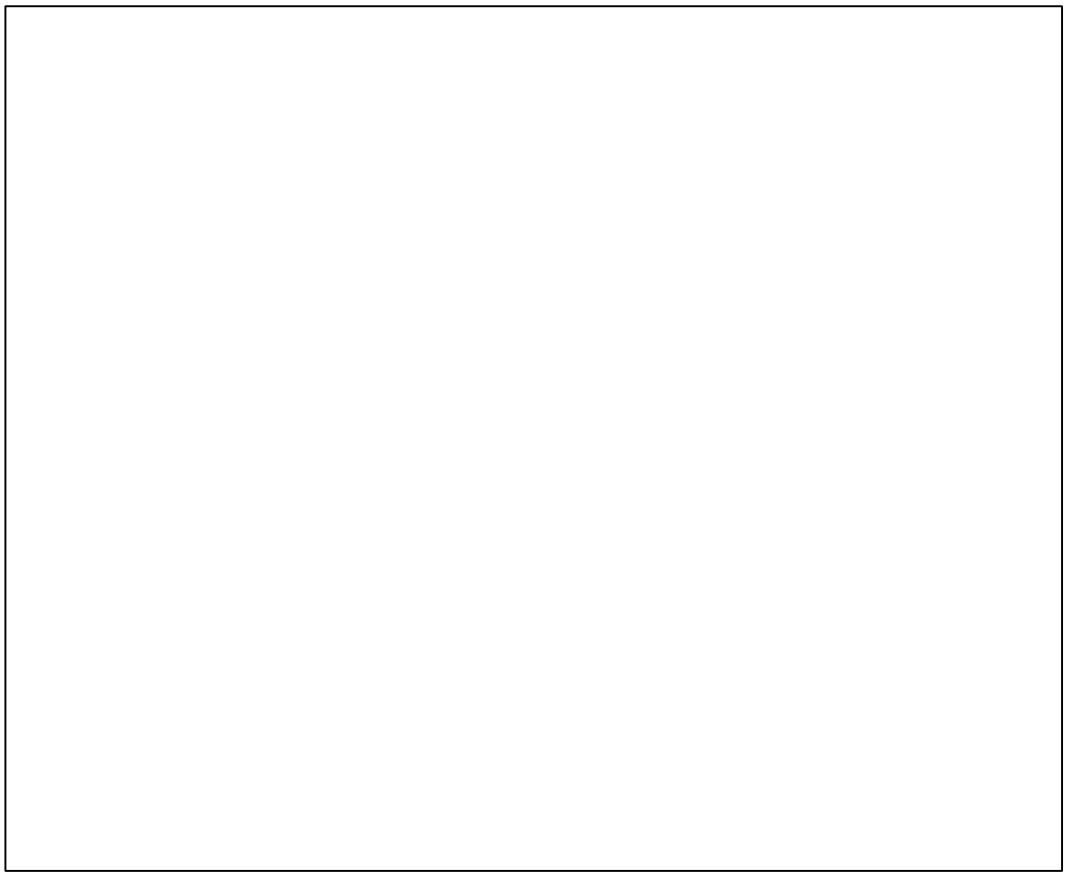
See [2] for the concise overall specification of LDAPv3.

Brief Review of Security Concepts

- Notion of *Security* for a network protocol is comprised of (at least) these axes..
  - **Authentication**
    - "Who are you and who says so?"
  - **Confidentiality**
    - "Tough petunias to eavesdroppers."
  - **Integrity**
    - "Did anyone muck with this data?"
  - **Authorization**
    - "Yes, you can do that, but no, you can't do that other thing."

  [23] provides a concise introduction to these concepts (plus lots more)

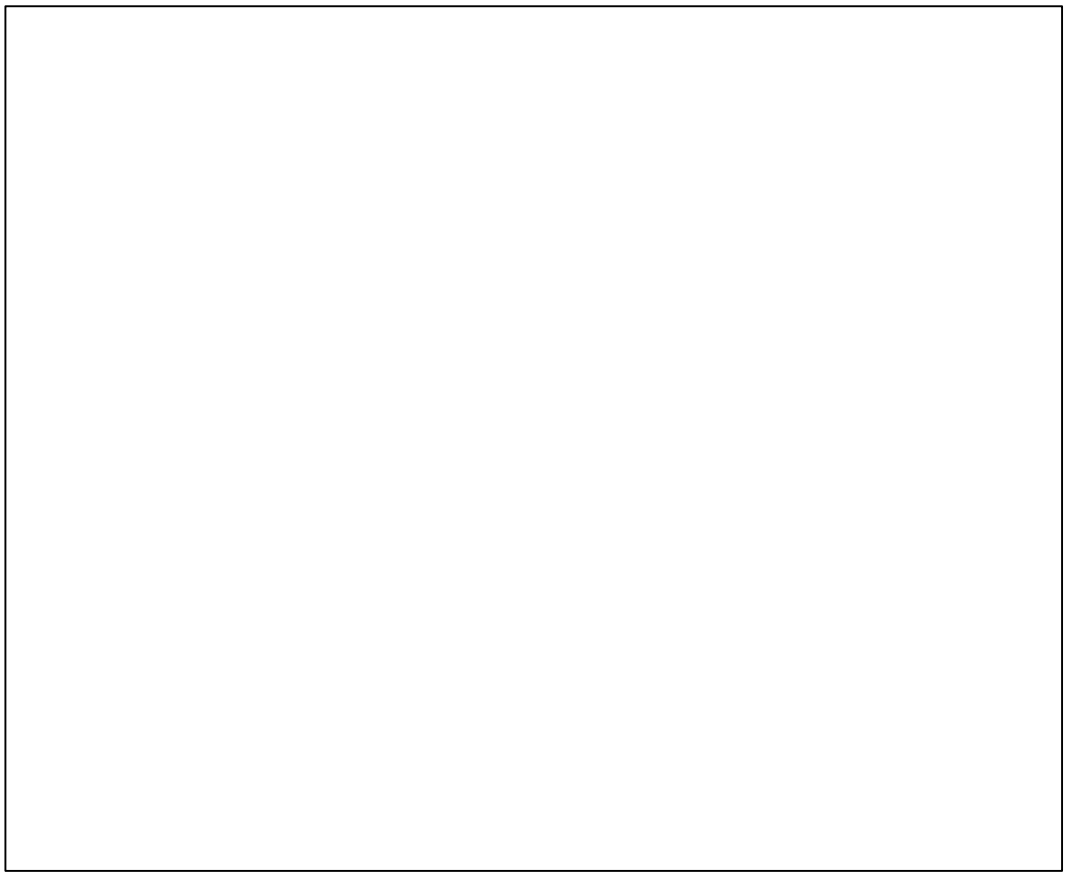
10-Apr-2001 Security Design for Application Protocols - Jeff Hodges 7



Brief Review of Security Concepts

- Additionally, the above four concepts
  - Authentication
  - Confidentiality
  - Integrity
  - Authorization
- ***ought*** to be applied to the ***data*** accessible via the protocol.
  - E.g. in the case of LDAP, applied to *directory data itself* [12]
    - E.g. DNSSEC (RFC 2535) embodies such functionality in the context of the Domain Name System (DNS).

10-Apr-2001 Security Design for Application Protocols - Jeff Hodges 8



Brief Review of Security Concepts

- The applicable "science & technology of implementation" are ciphers, hashes/digests, etc, and are used as tools to perform…
  - Encryption
  - Hashing
- In other words: ***Cryptography*** [17]
- *But*, there's far more to security than this…

10-Apr-2001 Security Design for Application Protocols - Jeff Hodges 9

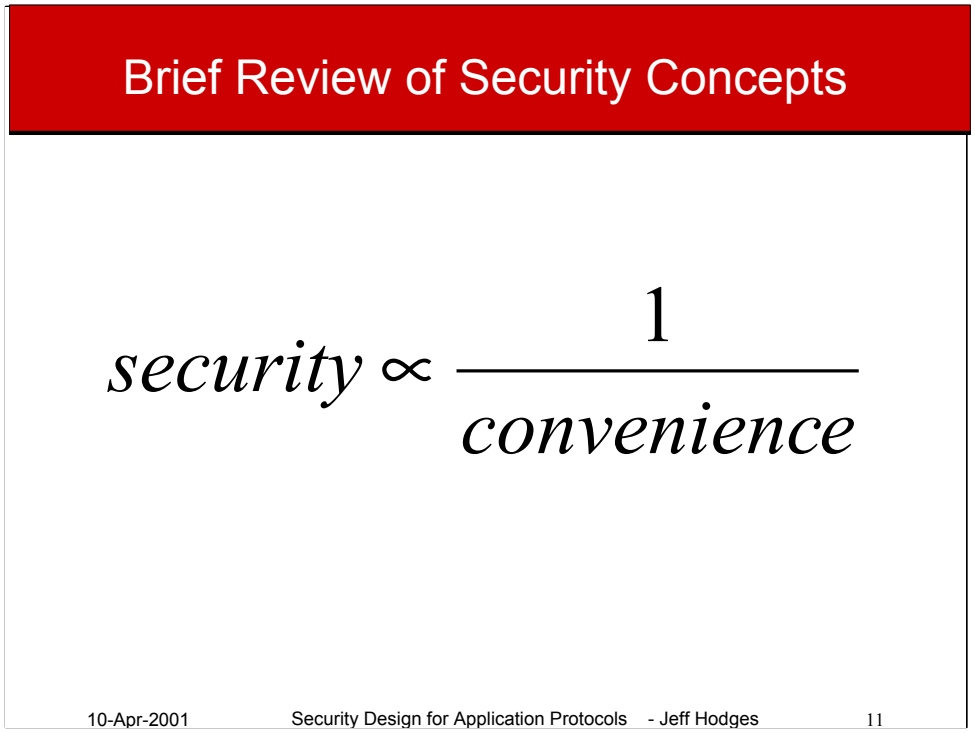
As illustrated in the following three slides…

..and this one.

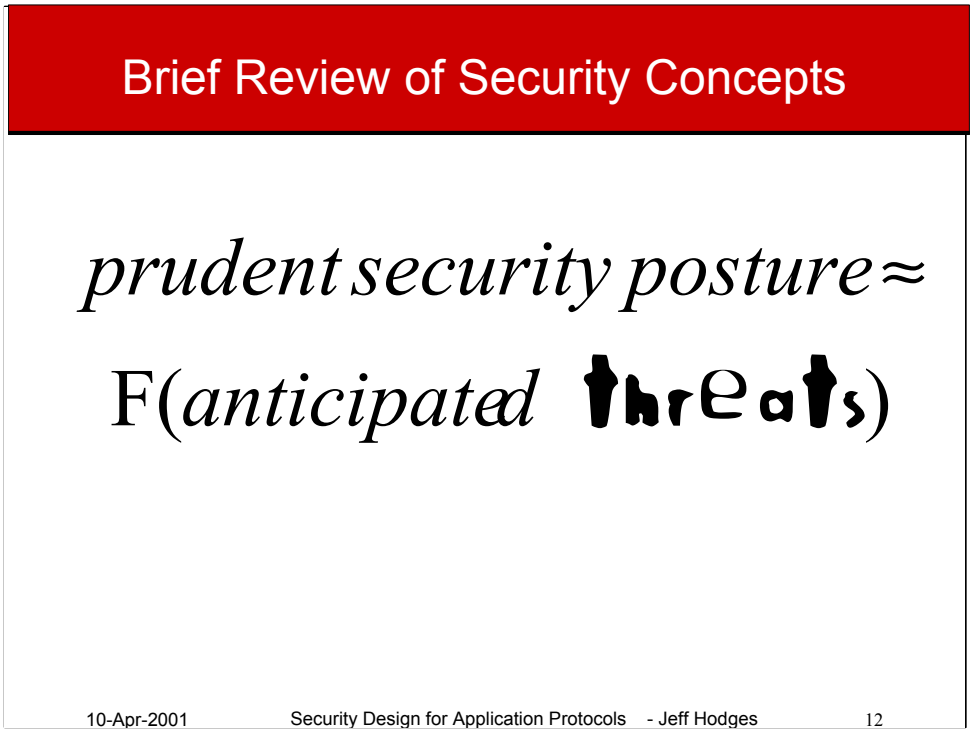
Refs: [11, 12, 14]

[above text fragment (that appears in the .pdf version of this talk) is irrelevant and appears to be a manifestation of some (annoying) Powerpoint bug. Sorry.]

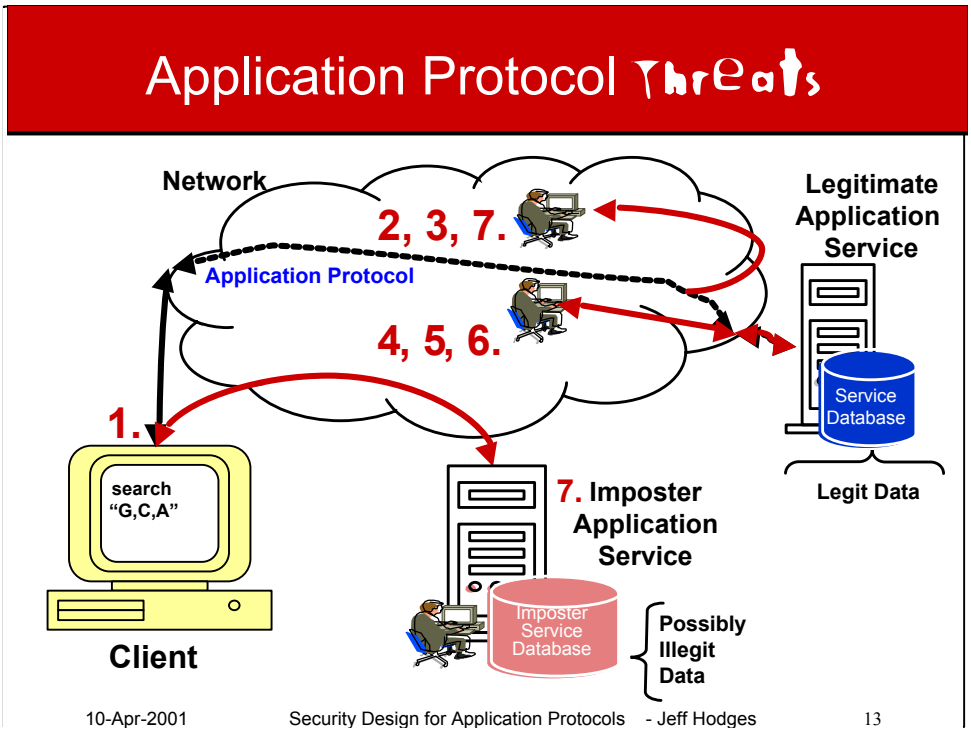
"It" – your protocol, application, overall system, and so on – is only as secure as the weakest link, in terms of design, implementation, and deployment. "Security is a process, not a product" [16]



Ensuring security often comes at the expense of convenience, one way or another. Which has a way of making otherwise legitimate users look sorta like the bad guys at times. [16, 19, 22]



Meanwhile, the level of security one is obliged to ensure (both in terms of design and deployment) should be driven by a function of the threats one anticipates, and one's tolerance for risk. [16, 18, 19, 20, 22, 23]



Application Service threats…

1. Unauthorized access to data via data-fetching operations,

2. Unauthorized access to reusable client authentication information by monitoring others' access,

3. Unauthorized access to data by monitoring others' access,

4. Unauthorized modification of data,

5. Unauthorized modification of configuration,

6. Unauthorized or excessive use of resources (denial of service), and

7. Service Impersonation: Tricking a client into believing that information came from the legitimate service when in fact it did not, either by modifying data in transit or misdirecting the client's connection.

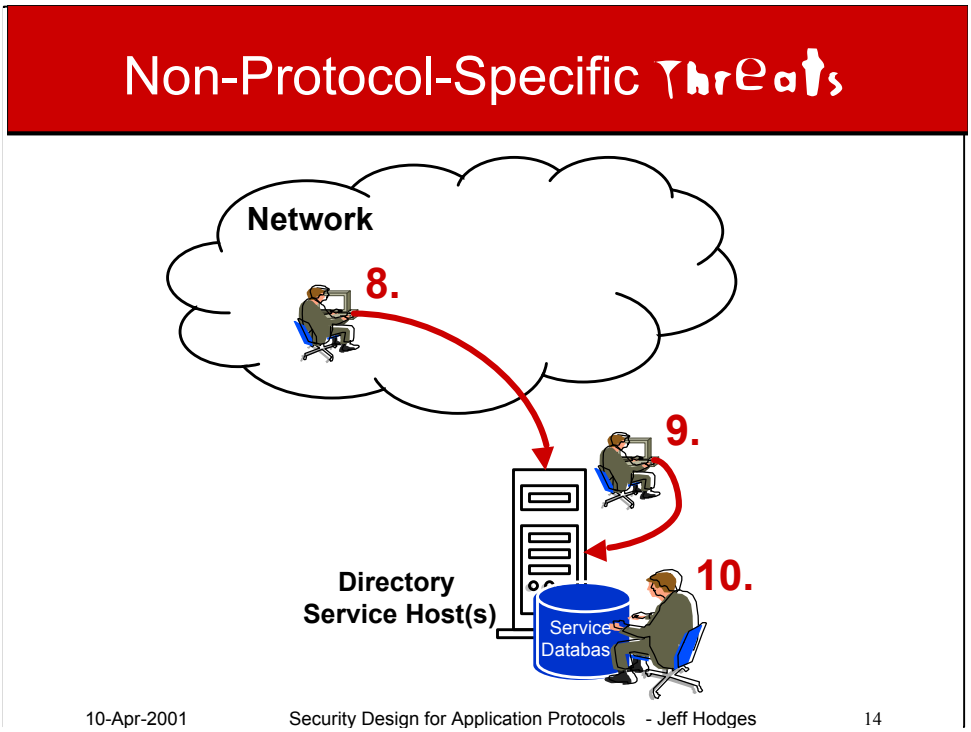
Myth..

Crackers-at-large are one's primary enemies.

Reality..

One's own administrators, employees, users are often a non-trivial source of threats, and should be considered right along with so-called external threats. See "Insiders versus Outsiders" sidebar on Page 112 of [22] <URL:http://www.nap.edu/readingroom/books/trust/trust-4.htm#Page 112>

Refs: [3, 4, 6, 7, 8, 10, 11, 16, 18, 20, 22, 23]



Plus, there's these deployment-specific (I.e. non-application-protocol-specific) threats..

8. Various network-based attacks against the application service hosts themselves -- e.g. against the OS, other network services running on the host, etc.

9. Various attacks against the host by someone with physical or near-physical access.

E.g. access to the system console,

access to a directly-connected serial line,

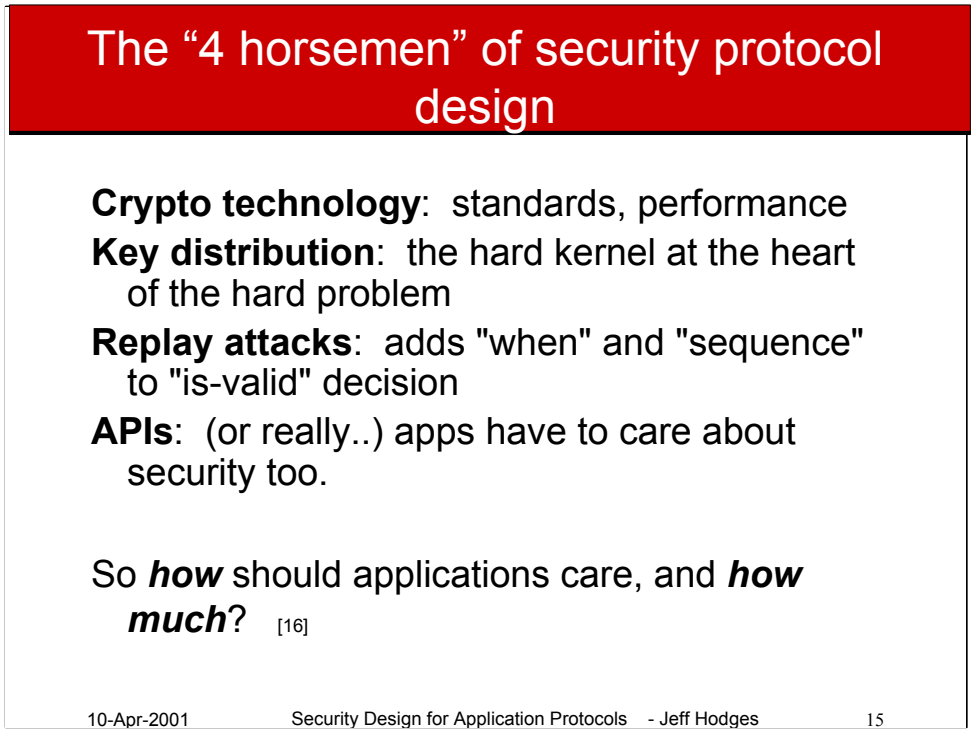
access to a directly-connected modem,

access to the system unit itself,

etc.

10. Attacks against the very media housing the directory database, e.g. simply stealing or copying the disk(s) itself.

Refs: [3, 11, 16, 18, 20, 22, 23]



..and this one.

Refs: [11, 12, 14]

[above text fragment (that appears in the .pdf version of this talk) is irrelevant and appears to be a manifestation of some (annoying) Powerpoint bug. Sorry.]

Security Mechanisms as a Function of threats & **Data** & Requesters

scenarios	Contains Sesitive Data?	Connection Hijacking or IP Spoofing Threats?	Anonymous Requesters?		Identified Requesters?		Prudent Security Mechanisms or Functions
				Read/Write ?		Read/Write ?	
1	N	N	**Y**	RO	N		None
2	N	N	N	N/A	**Y**	RO	Reqstr **Authentication**
3	N	**Y**	N/A	N/A	N/A	N/A	Mutual **authentication**, Connection **Integrity-Protection**
4	N	N	Y	RO	Y	RW	Reqstr **Authentication**
5	**Y**	**Y**	N/A	N/A	N/A	N/A	Mutual **authentication,** Connection **Integrity-** and **Confidentiality-**

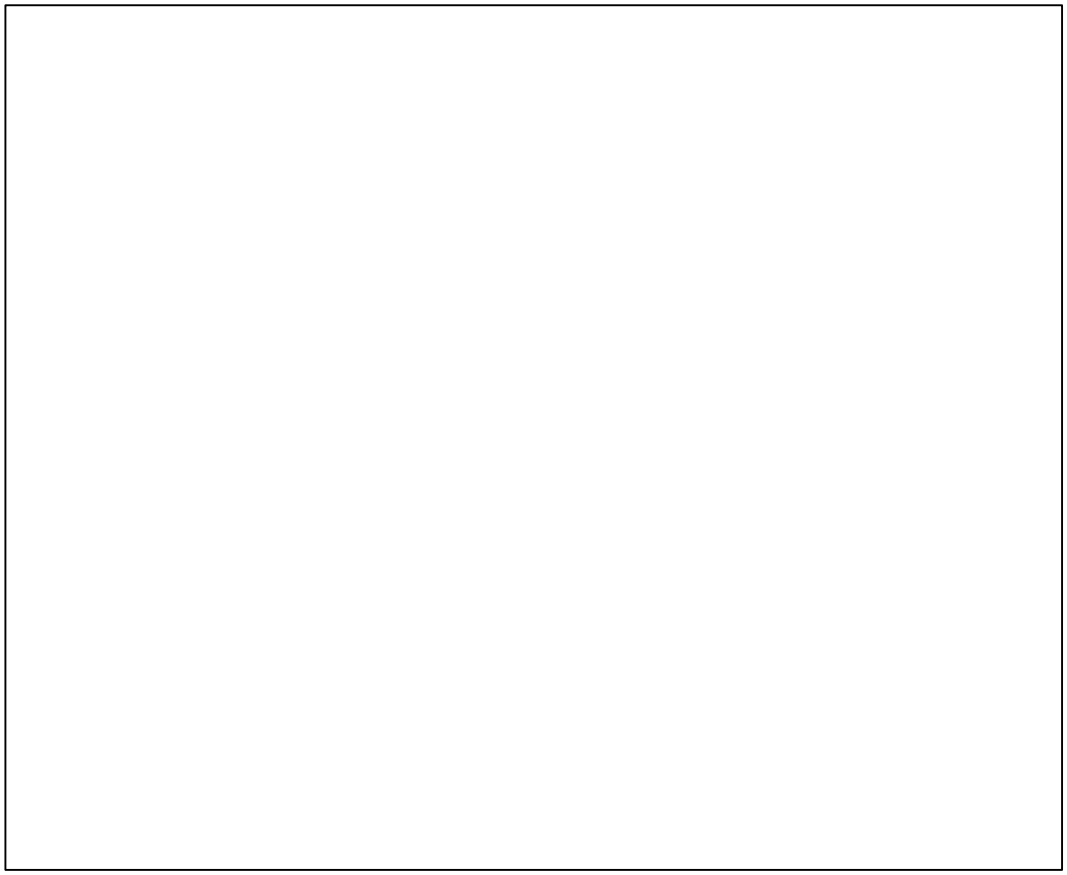
10-Apr-2001 Security Design for Application Protocols - Jeff Hodges 16

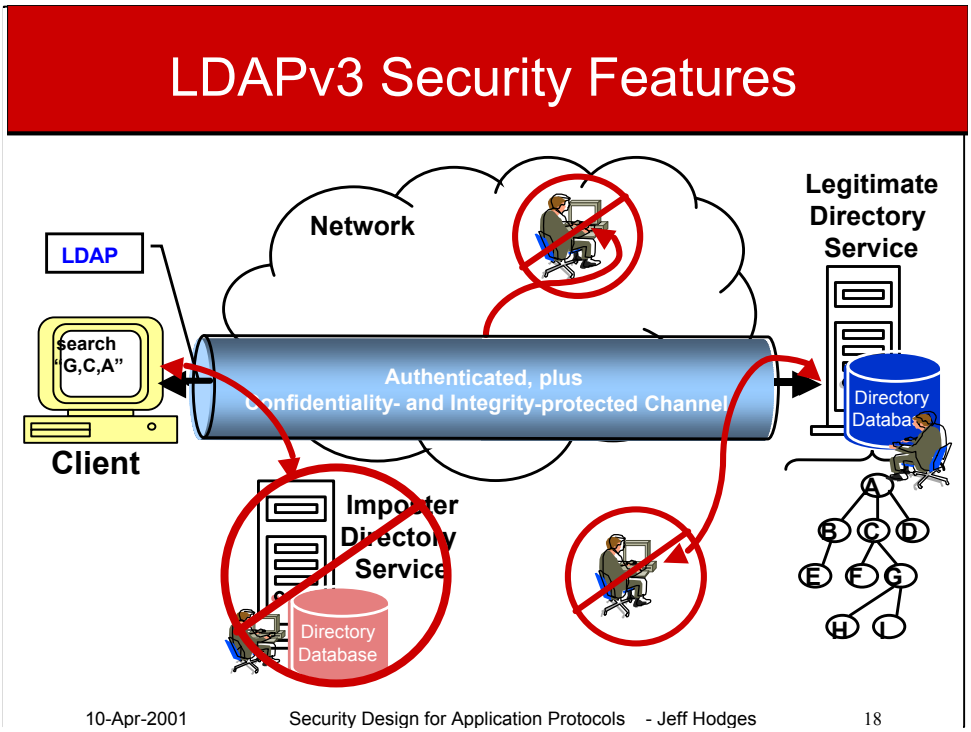
This is an example of the level of thinking & caring about security we did for LDAPv3 [3]. However, it is essentially applicable to most any application-layer protocol. This table would apply to any information-retrieval app protocol built using the BEEP [1] framework, for example. Or on top of HTTP, SOAP, et al.

Source: [3] “Authentication Methods for LDAP”, RFC 2829 (aka “AuthMeth”), Section 4. Note that there certainly are other valid combinations -- this table (and that section of AuthMeth) isn't intended to be exhaustive.

Example: LDAPv3 Protocol Security Features

- Formal notions of..
  - Authentication Identifiers, and..
  - Authorization Identifiers (see: [3, 4, 5, 8, 9])
- Leverages several security mechanisms..
  - Simple passwords [3, 6, 14]
  - SASL [8]
    - Kerberos [7]
    - Digest [6]
  - SSL/TLS [4, 5, 9]
    - effectively is a *session layer*
- The above may be used in various combinations together. [3, 4, 5]



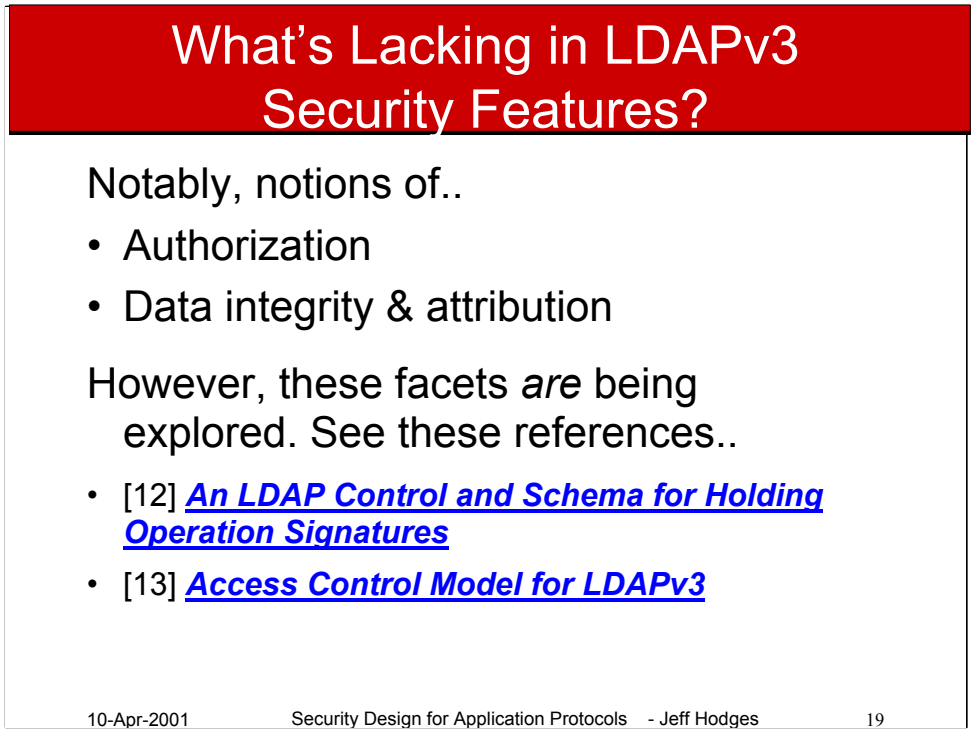


The network-based threats against the directory protocol are *largely attenuated* by having strong authentication, and a security layer.

Note that the this illustration also applies to a app protocol built using the BEEP framework.

Though, note that threats 8, 9, 10 (illustrated on slide 14) are still issues. But to what extent one attempts to address them in practice is determined by anticipated threats and one's tolerance for *risk*.

See [11, 17, 19, 21, 22, 23] for more info, and also to shed light on why the network-based threats are likely only "largely attenuated" rather than being "decisively eliminated".

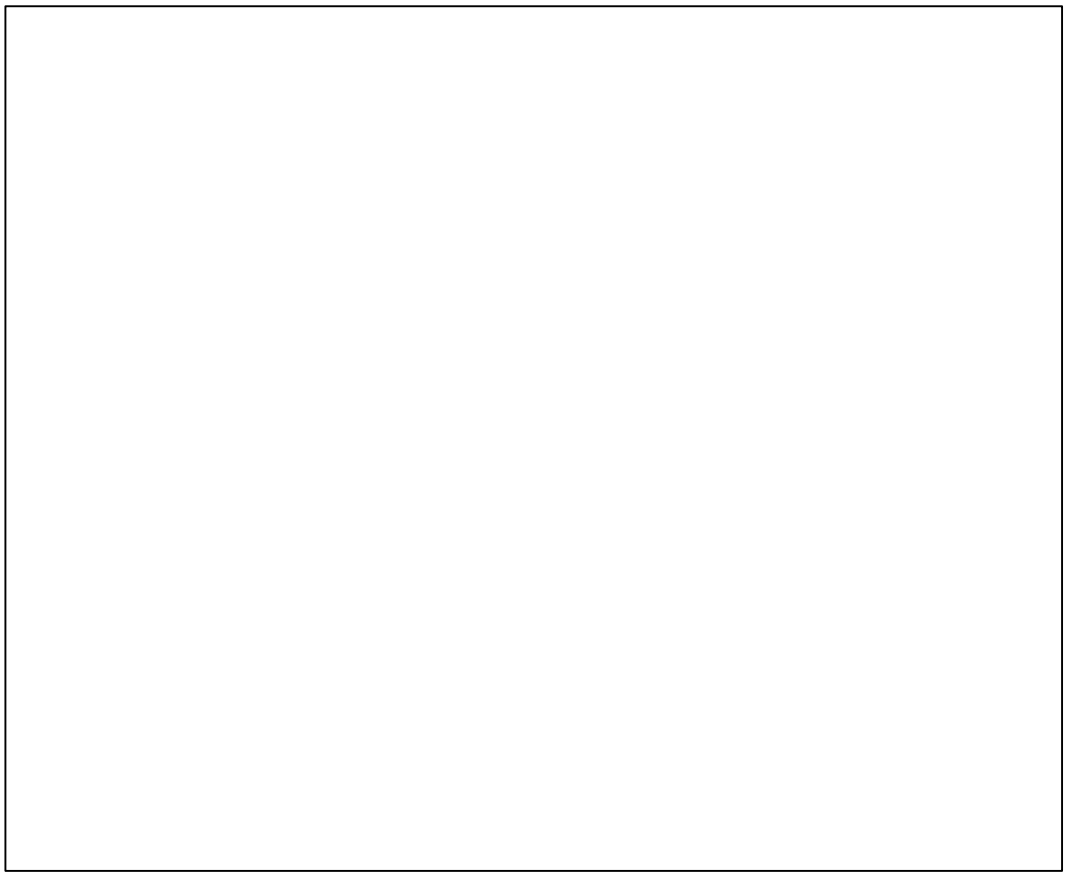


Note that [12, 13] are works-in-progress.

Decent Examples, again

- LDAPv3
  - See especially RFCs 2829 & 2830 [3, 4]
- BEEP
  - RFC 3080 [1]
- Others? HTTP (RFC 2817)?

Amongst this, there's an important point to consider…



If you're going to design an app protocol…

- …and you use the BEEP framework, then you automatically have much of the new protocol's security considerations addressed.

The End

…
… hmm….
…..wellll….
…uh…hmmm….

…….uhm… Is this really “the end” of this story? Aren’t we neglecting other aspects, such as protecting against stuff like…

Denial of Service ?

Well, yes. But there’s no pre-packaged tools or techniques that presently address such stuff (especially at the application layer) so it is unfortunately out-of-scope for this talk. But, it is ***not*** out-of-scope for the IETF et al to be thinking about.

References

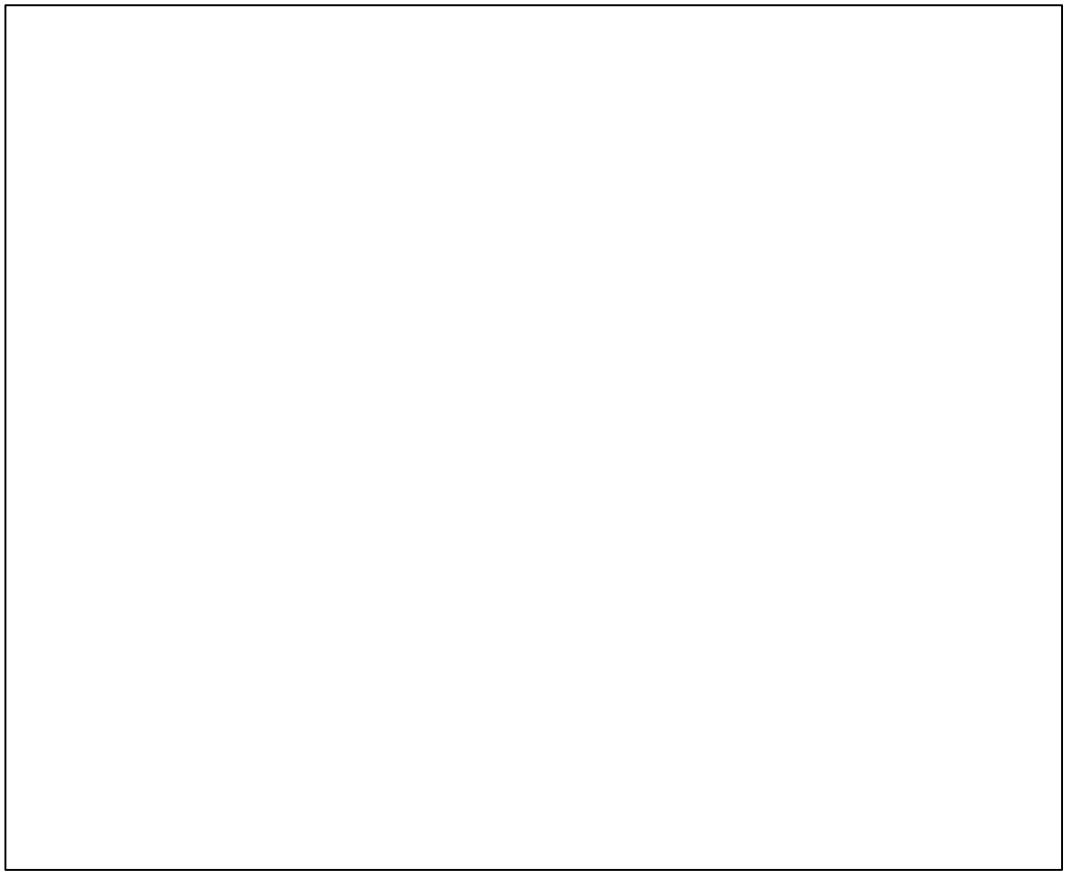
- This talk is available at..
  - **http://www.stanford.edu/people/hodges/talks/**
- References..
  - [1] ***The Blocks Extensible Exchange Protocol Core***. Marshall Rose. RFC 3080, March 2001.
    Available at: http://www.ietf.org/rfc/rfc3080.txt
  - [2] ***Lightweight Directory Access Protocol (v3): Technical Specification***. J. Hodges & RL "Bob" Morgan, INTERNET-DRAFT work-in-progress, 12 January, 2001.
    Available at: http://www.ietf.org/internet-drafts/draft-ietf-ldapbis-ldapv3-ts-00.txt
  - [3] ***Authentication Methods for LDAP***. M. Wahl, H. Alvestrand, J. Hodges, R. Morgan. RFC 2829, May 2000.
    Available at: http://www.ietf.org/rfc/rfc2829.txt

References, cont'd

- [4] ***Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security***. J. Hodges, R. Morgan, M. Wahl. RFC 2830, May 2000.
  Available at: http://www.ietf.org/rfc/rfc2830.txt
- [5] ***LDAP Association State Diagram***. Jeff Hodges, Revision 2.0b, December 1999.
  Available at: http://www.stanford.edu/~hodges/doc/LDAPAssociationStateDiagram-1999-12-14.html
- [6] ***Digest Authentication as a SASL Mechanism***. P. Leach, C. Newman. RFC 2831, May 2000.
  Available at: http://www.ietf.org/rfc/rfc2831.txt
- [7] ***The Kerberos Network Authentication Service (V5).*** J. Kohl, C. Neuman. IETF Request For Comments RFC 1510, September 1993.
  Available at: http://www.ietf.org/rfc/rfc1510.txt
- [8] ***Simple Authentication and Security Layer (SASL).*** J. Myers. IETF Request For Comments RFC 2222, October 1997.
  Available at: http://www.ietf.org/rfc/rfc2222.txt

References, cont'd

- [9] ***The TLS Protocol Version 1.0***. T. Dierks, C. Allen. RFC 2246, January 1999.
  Available at: http://www.ietf.org/rfc/rfc2246.txt
- [10] ***IP Security: Document Roadmap***. R. Thayer, N. Doraswany, R. Glenn. IETF Request For Comments RFC2411, November 1998.
  Available at: http://www.ietf.org/rfc/rfc2411.txt
- [11] ***Site Security Handbook***. B. Fraser, Editor. IETF Request For Comments RFC2196, FYI8. September 1997.
  Available at: http://www.ietf.org/rfc/rfc2196.txt
- [12] ***An LDAP Control and Schema for Holding Operation Signatures***. B. Greenblatt, P. Richard. IETF Request For Comments 2649 [Experimental], August 1999.
  Available at: http://www.ietf.org/rfc/rfc2649.txt
- [13] ***Access Control Model for LDAPv3***. E. Stokes, B. Blakley, D. Rinkevich, R. Byrne. INTERNET-DRAFT work-in-progress, 2 March 2001.
  Available at: http://www.ietf.org/internet-drafts/draft-ietf-ldapext-acl-model-07.txt

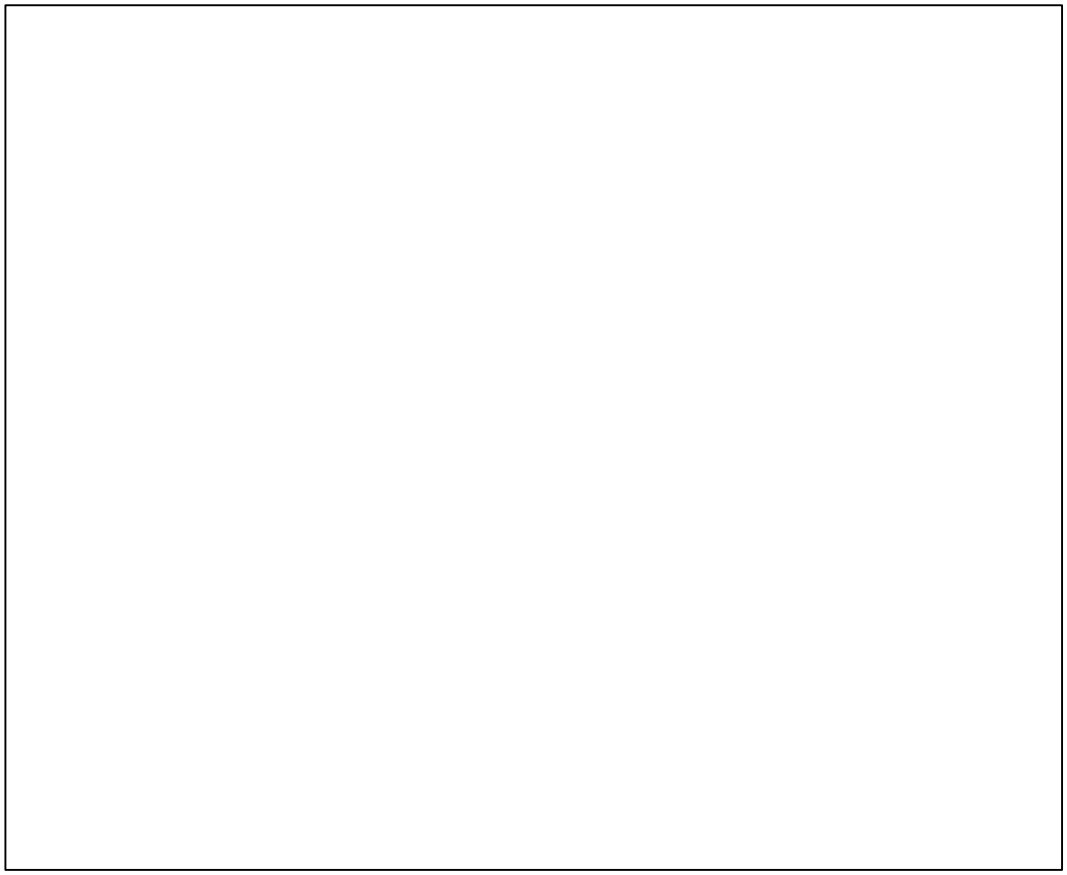


References, cont'd

- [14] ***Lightweight Directory Access Protocol (v3)***. M. Wahl, T. Howes, S. Kille, IETF Request For Comments RFC2251, December 1997.
  Available at: http://www.ietf.org/rfc/rfc2251.txt
- [15] RL "Bob" Morgan, personal communication, based on his notes from Jeff Schiller's pre-IETF Security Tutorial at the 49th IETF in San Diego, Dec-2000.

- Security books, papers, etc (in no particular order)…

- [16] ***Secrets and Lies***. Bruce Schneier, John Wiley & Sons, Inc., 2000. ISBN: 0471253111
- [17] ***Applied Cryptography - Protocols, Algorithms, and Source Code in C (Second Edition).*** Bruce Schneier, John Wiley & Sons, Inc., 1996. ISBN: 0471117099.



References, cont'd

- [18] ***Practical UNIX & Internet Security, 2nd Edition***. Simson Garfinkel and Gene Spafford, O'Reilly & Associates, April 1996, ISBN: 1-56592-148-8.
- [19] ***Risk Management is Where the Money Is***. Dan Geer, CertCo, November 1998. Available at: http://catless.ncl.ac.uk/Risks/20.06.html#subj1.1
- [20] ***Web Security & Commerce***. Simson Garfinkel with Gene Spafford, O'Reilly & Associates, June 1997, ISBN 1-56592-269-7.
- [21] ***Why Cryptography Is Harder Than It Looks***. Bruce Schneier, Counterpane Systems, 1996.
- [22] ***Trust in Cyberspace***. Committee on Information Systems Trustworthiness, Fred B. Schneider - Editor, National Research Council, ISBN 0-309-06558-5, 1999. On-line copy and ordering information available at: http://www.nap.edu/readingroom/books/trust/
- [23] ***Network Security Essentials: Applications and Standards***. William Stallings, Prentice Hall, 2000, ISBN: 0-13-016093-8.