IT Security:
New Trends, Ancient Techniques

Alec Muffett

Chief Architect, Security

Sun Professional Services EMEA

Sun microsystems

We make the net work.

Alec Muffett: Work

- SunPS Europe, Mid-East & Africa
  - Chief Architect for Security, EMEA
  - 18+ years of Network Security
  - 12+ years for Sun Microsystems
- Specialising in:
  - Network Architecture
  - Network Security Auditing
  - Authentication & Cryptography

Alec Muffett: Leisure

- Open Source Author
  - “Crack” Password Cracker
  - “CrackLib” Password Integrity Checker
  - Original USENET Security FAQ
  - Various other security stuff
- RSA Factorisation
  - BlackNet: 384-bit Secret/PGP key break
  - RSA155: world's first 512bit RSA break

Proposition

- There are many security “components” available today…
  - Components == Tools, Utilities, Appliances…
  - Available across many platforms…
  - Addressing many specific security risks…
  - Multi-billion dollar industry
  - But…

Proposition

- Many useful security “components” are available, but…
  - They are easily misassembled, misused or misconfigured
  - They are often better used in combination, rather than individually

Further...

- Regarding “components”
  - People tend to seek “homogeneity”, whereas “diversity” yields greater robustness, at the cost of management complexity
  - Without proper design/architecture, you will be wasting your money

    It is perfectly possible to spend $1,000,000's and yet have a terribly insecure network...

    It is further possible to spend almost nothing, and yet improve your security enormously.

Therefore, today...

- We shall:
  - review what may go wrong.
  - review how it can go wrong.
  - suggest a strategy, even a design philosophy which helps to address it.

As a first illustration…

- Consider this problem:
  - The misassembly of security components
  - Viz: “Right Components, Put Together Wrong”
  - Example:

    TopBox Breaking Video (2m 50s)

So Let's...

- Recap some history of IT Security
- Compare it to the “State of the Art”
- Review individual security tools, and the issues they sought to address
- Determine whether we are still defending against the “older” security issues

IT Security –
A Rough History

IT Security Eras

- 1955..65
  - Computers “too complex for ordinary people” yielding “security through obscurity”

IT Security Eras

- 1965..75
  - Advisory separation of different “users” within a computer
  - Technology (mostly) not advanced enough to support mandatory separation.
    - Lack of VM, etc

IT Security Eras

- 1975..85
  - Partitioning of file access via robust “file permissions”
  - Strong “virtual memory” to enforce mandatory user/program separation...
  - ...but not in all platforms (eg: Personal Computers)

IT Security Eras

- 1985..95
  - Password security extended to basic network services
  - Networking “too complex for ordinary people” yielding “security through obscurity” again
  - ...yet early “buffer-overflow” exploits occur
  - Personal Computers virus-ridden from lack of technology to implement “integrity”
  - Compartmented/Certified Systems considered “exotic”; Military & Banking only?

IT Security Eras

- 1995..now
  - Partitioning of service access via firewalls
    - Firewalling used as panacea
    - Impact upon network architecture and throughput
  - Personal Computers begin to employ strong permissions, VM (etc) to ensure integrity...
  - ...boosting subsequent growth in “macroviruses” and “active-content” exploits in popular applications, to fill the gap.

IT Security Future?

- 2005+ …
  - What is the next big, open resource that is fit to protect with mandatory controls?
    - Encapsulated Data Security / Per-Object Crypto ?
    - Proximity wireless / Bluetooth?
    - SMS-Firewalling & Antivirus?
    - Your guess is as good as mine…

Implementation Cycles

- Generalising:
  - New resource/tool becomes available
    Identity, Filestore, Network, E-mail...
  - Resource/tool grows in popularity
  - Access restriction to/by the resource is layered-on afterwards
    Passwords, Permissions, Firewalls, Virus scanner...

Security Deployment

- Problem:
  - Access controls which are designed "after the fact" are often sub-optimal
    - Eg: Password protection on plaintext HTTP
    - Eg: Session-State Cookies in HTTP
    - Eg: 40-bit WEP in 802.11b
  - Arguably all of the above could have been forseen and implemented "properly"

Security Deployment

- In security, often only the latest "trendy" issues are managed…
- …to the detriment of others.
  - Weak file permissions on a big server
  - Ignored because:
    "The firewall does all our security!!!"
  - How many people here have hardened every server they own?

So Why Do Security?

- What are we protecting?
  - Data has value to us, and to “others”.
  - Data is valuable but intrinsically defenceless.
  - Data exists in more places for shorter or longer periods of time – caches, routers; how many of these places do you actually own?
- How shall we protect it?
  - So what we actually do to protect that which we value?

Issues of Implementation

- We actually protect the containers where data exists!
  - But: data exists in many places!
  - Hence the need to defend:

    Multiple data containers
    In multiple places
    At the same time.
  - This explains why security is “complicated”

Same Problems, Repeating

- Rough Categories of Challenge:
  - Over-reliance on one security technology
  - Blithe trust in what you are told
  - Reusable weak authentication
  - The right tools, put together wrongly

Same Problems, Repeating

- Overreliance upon single technologies
  - Obscurity
  - Permissions
  - Passwords
  - Firewalls
  - IDS
  - For instance:

    Potato famine

    Antibiotic Resistance

Same Problems, Repeating

- Blithe trust in what you are told
  - Unauthenticated identity
  - Buffer overflows (sometimes)
  - WWW Cookies
  - For instance:

    Forged passports / identity papers

    Social engineering

Same Problems, Repeating

- Reusable weak authentication
  - Plaintext passwords
  - Unencrypted Communications
  - Compare:

    Story of “Ali-Baba and the 40 Thieves”
    Reusable Password: “Open Sesame”
    Published circa 950AD
    A 1000-year-old IT security issue!

Same Problems, Repeating

- Right Tools, Assembled Wrongly
  - Firewalls with far too many “holes”
  - Firewalls with too much complexity
  - Same firewall technology everywhere
  - Poor Network Design

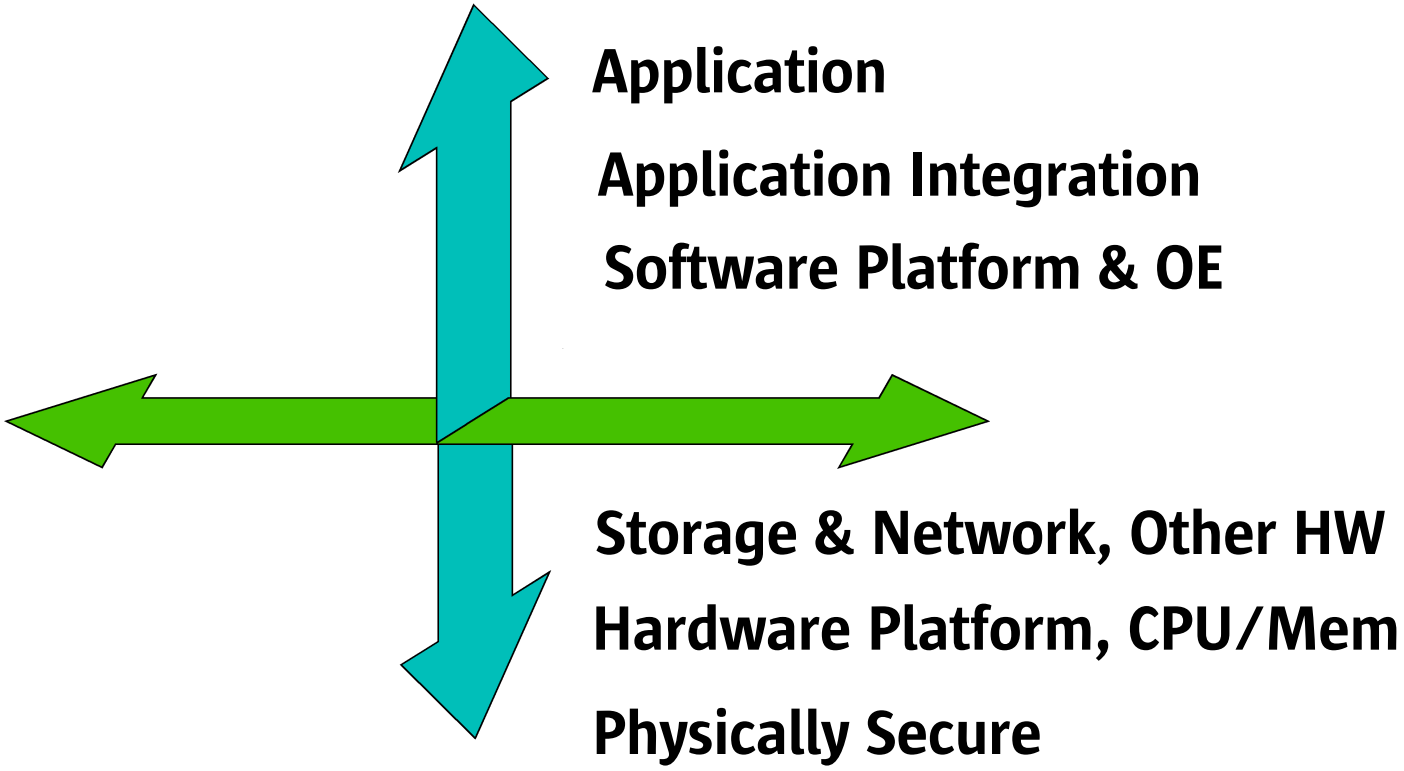
    Example: Simple SSL Accelerator (later…)

Attempts to Address Security: “End To End” Security

End-to-End: Communication

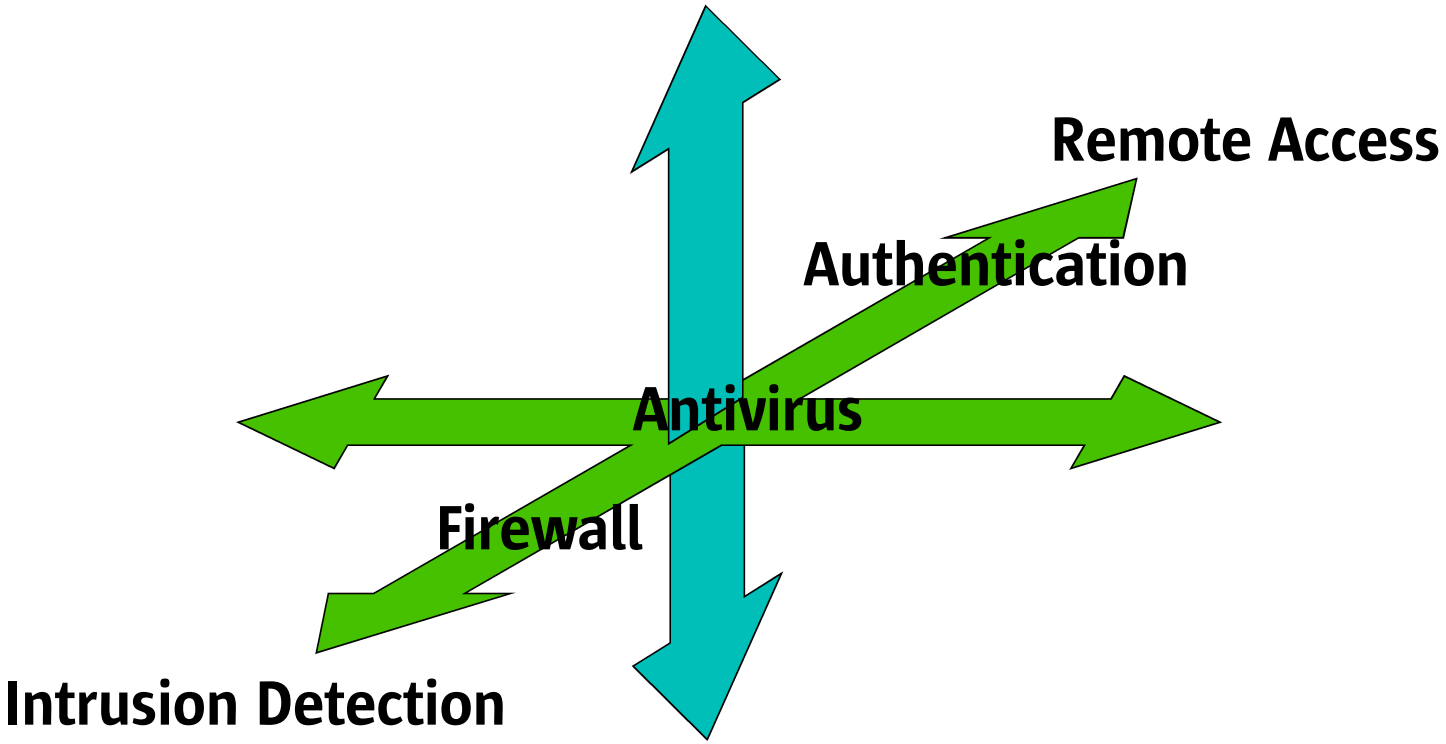
Secure & Authenticated Communication

End-to-End: Integration



Proper Integration of Components

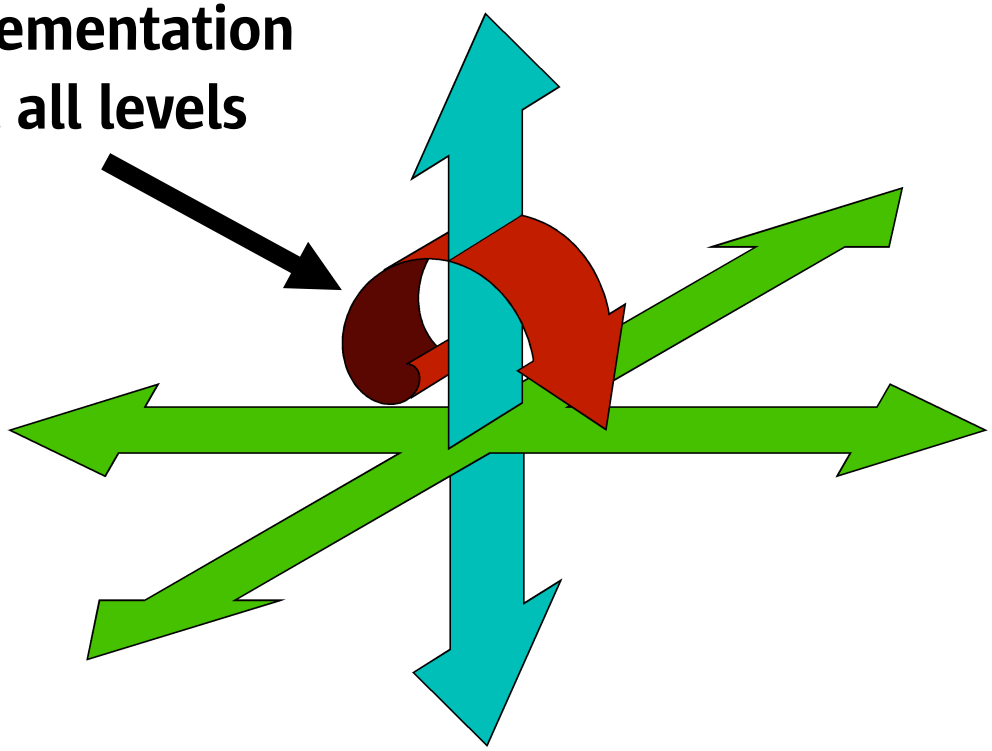
End-to-End: Functionality



Spectrum of Security Functionality

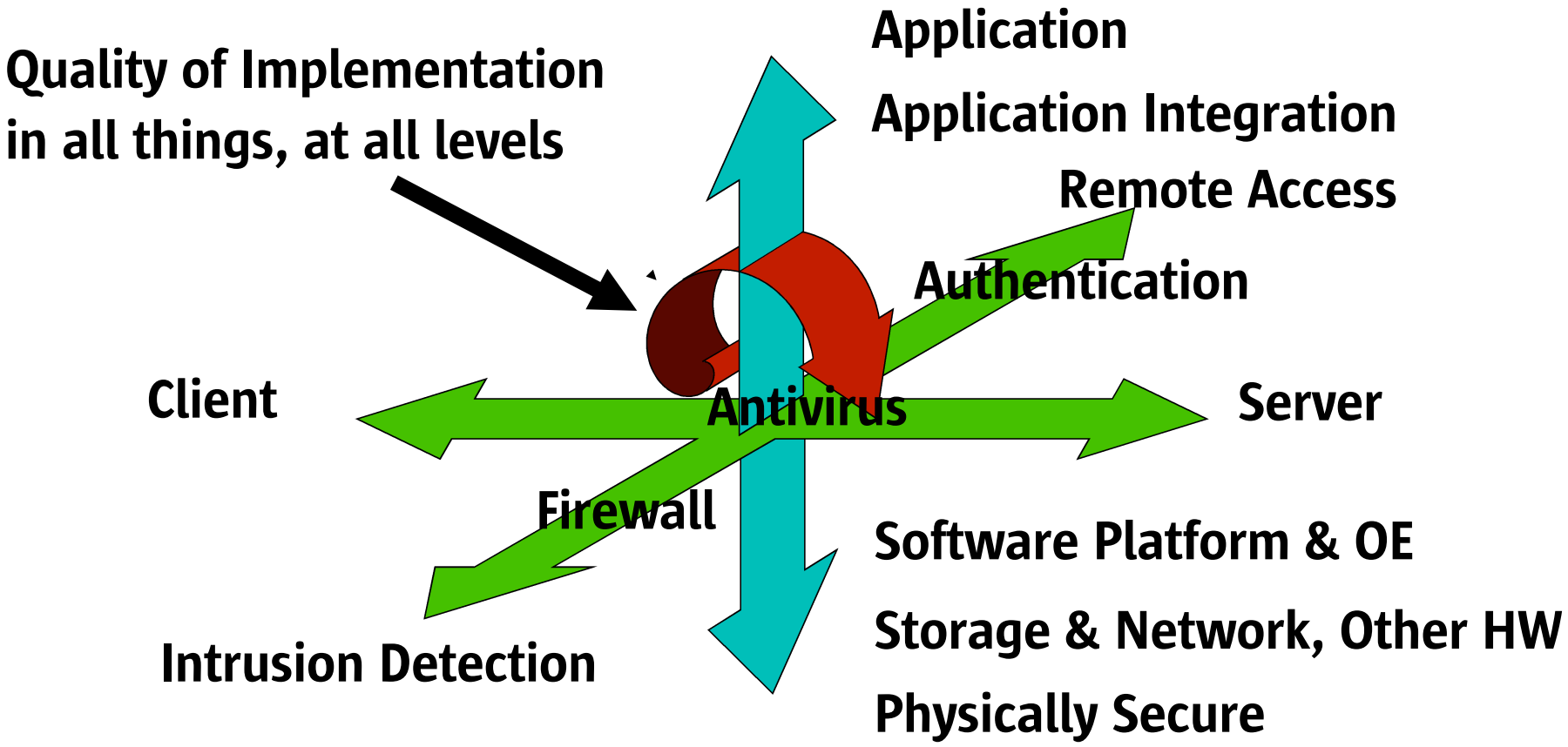
End-to-End: Quality

Quality of Implementation in all things, at all levels



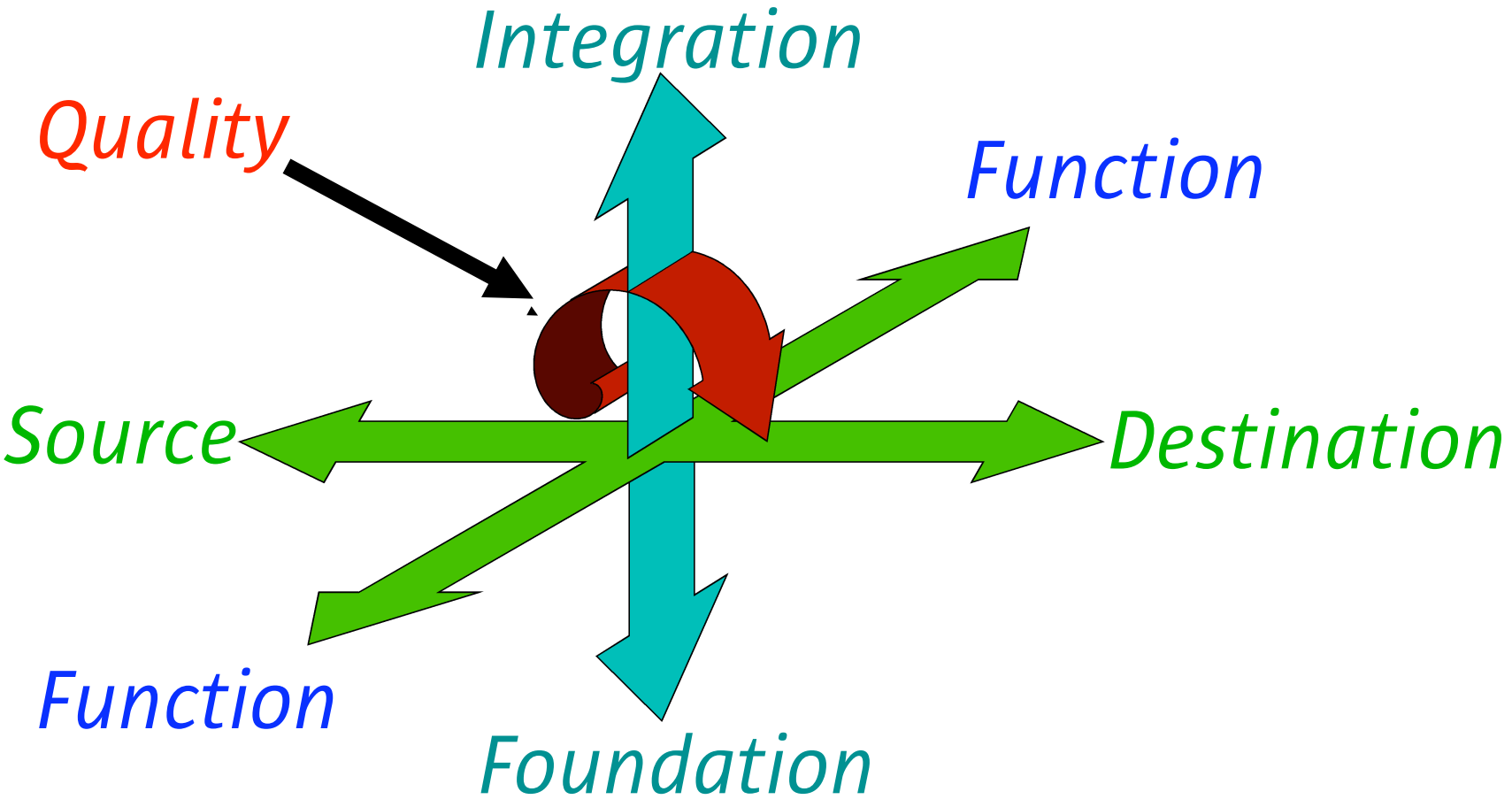
Quality of Components

End-to-End: 4-D Security



Apparently Confusing, but...

End-to-End: Simple 4-D Security



...actually rather simple.

When 2-D Drawings Fail...

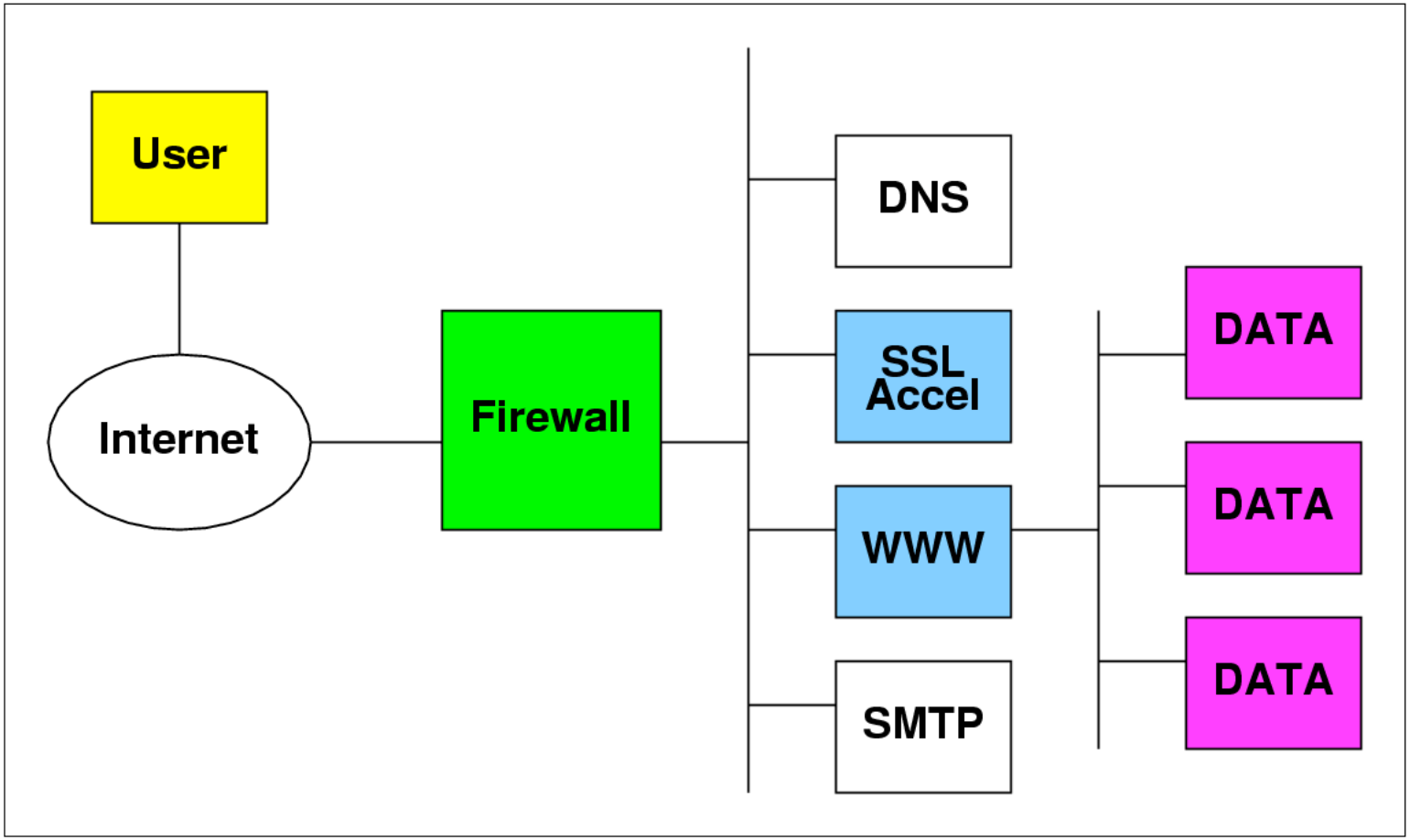
There is even a fifth, "*Human*" dimension to security, that which pertains to having correct "*People, Policy and Procedure*" - there are probably more.

Now consider:

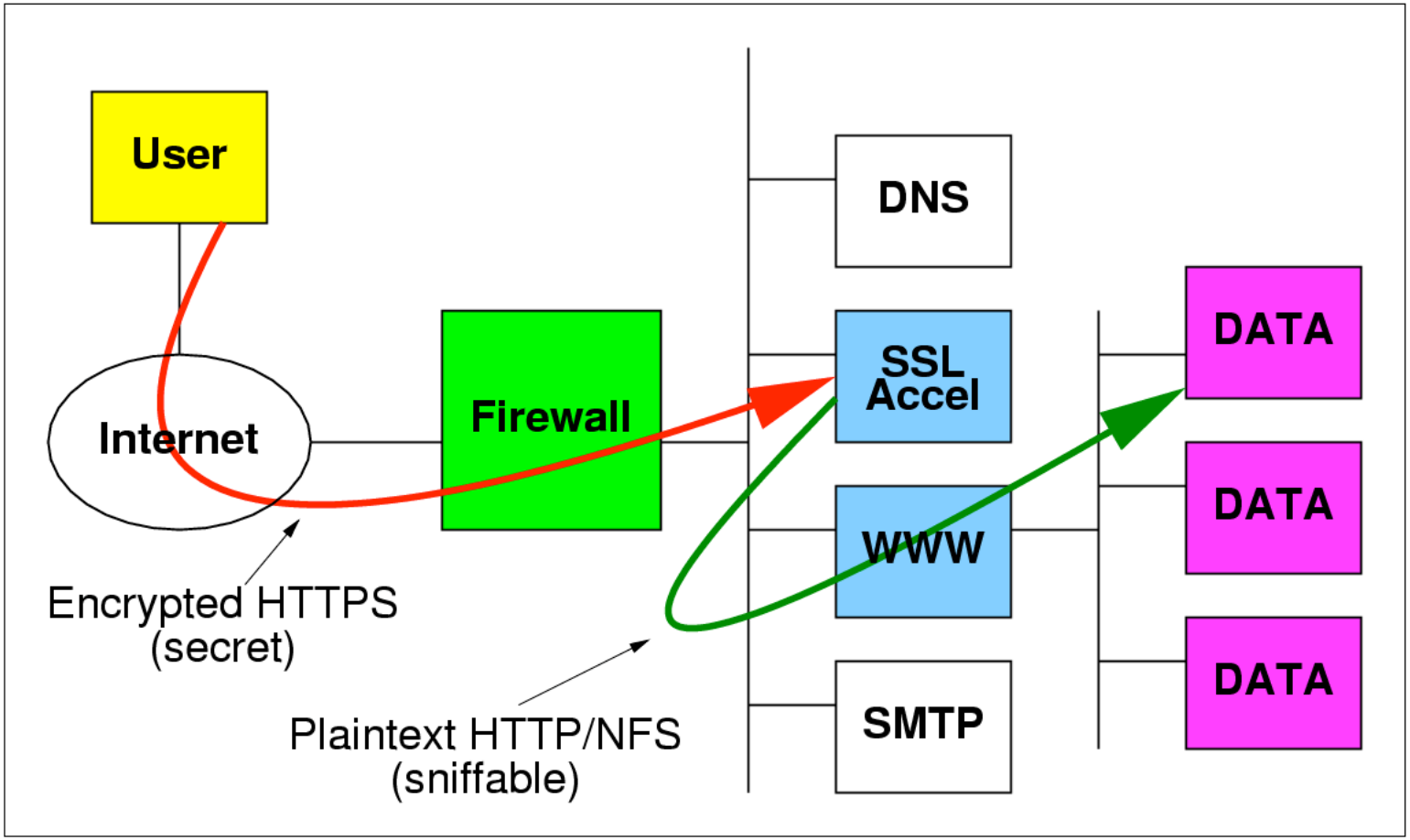
Does *your* security solution address all of these *dimensions*?

Better Security Through Better Design

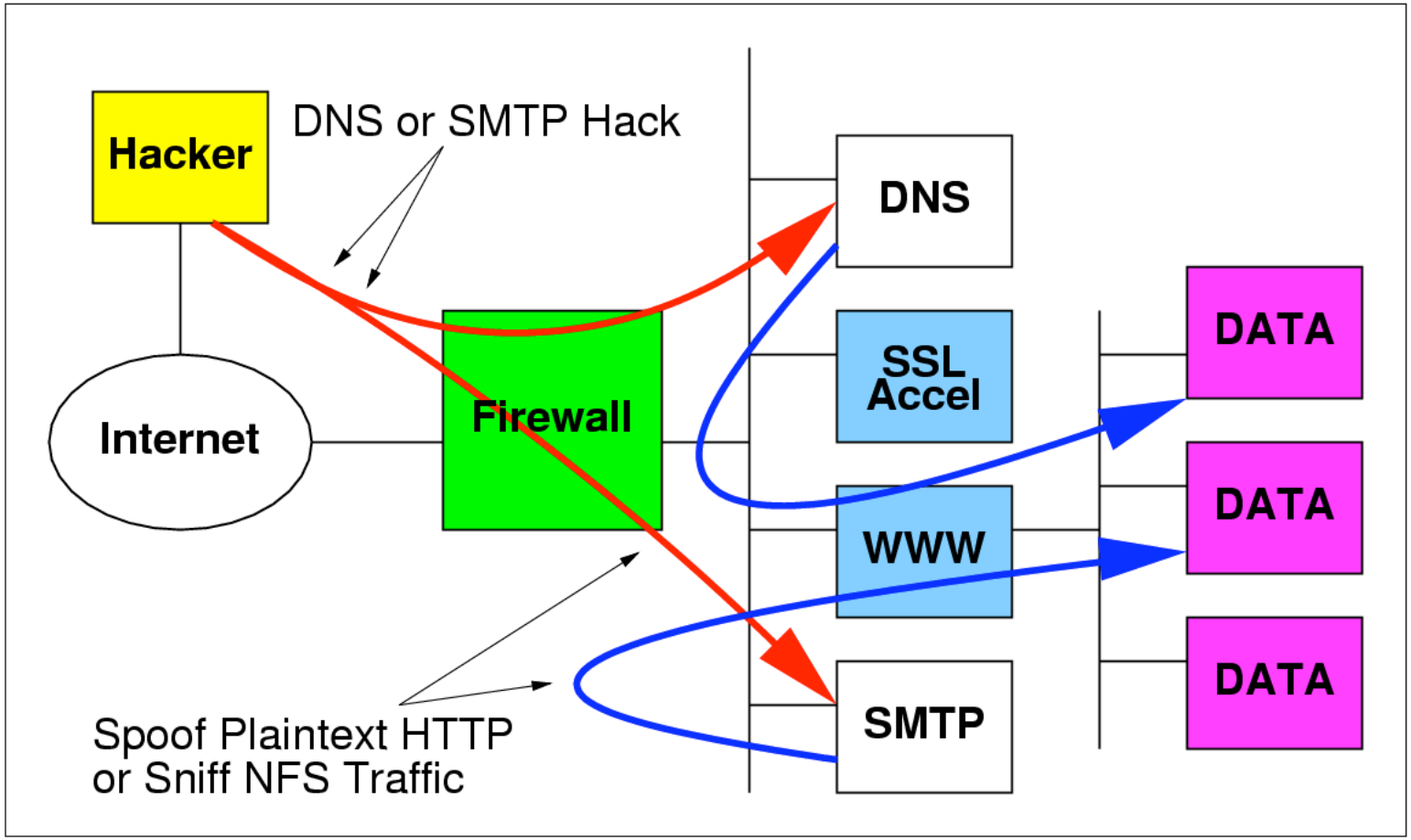
Design Example: SSL



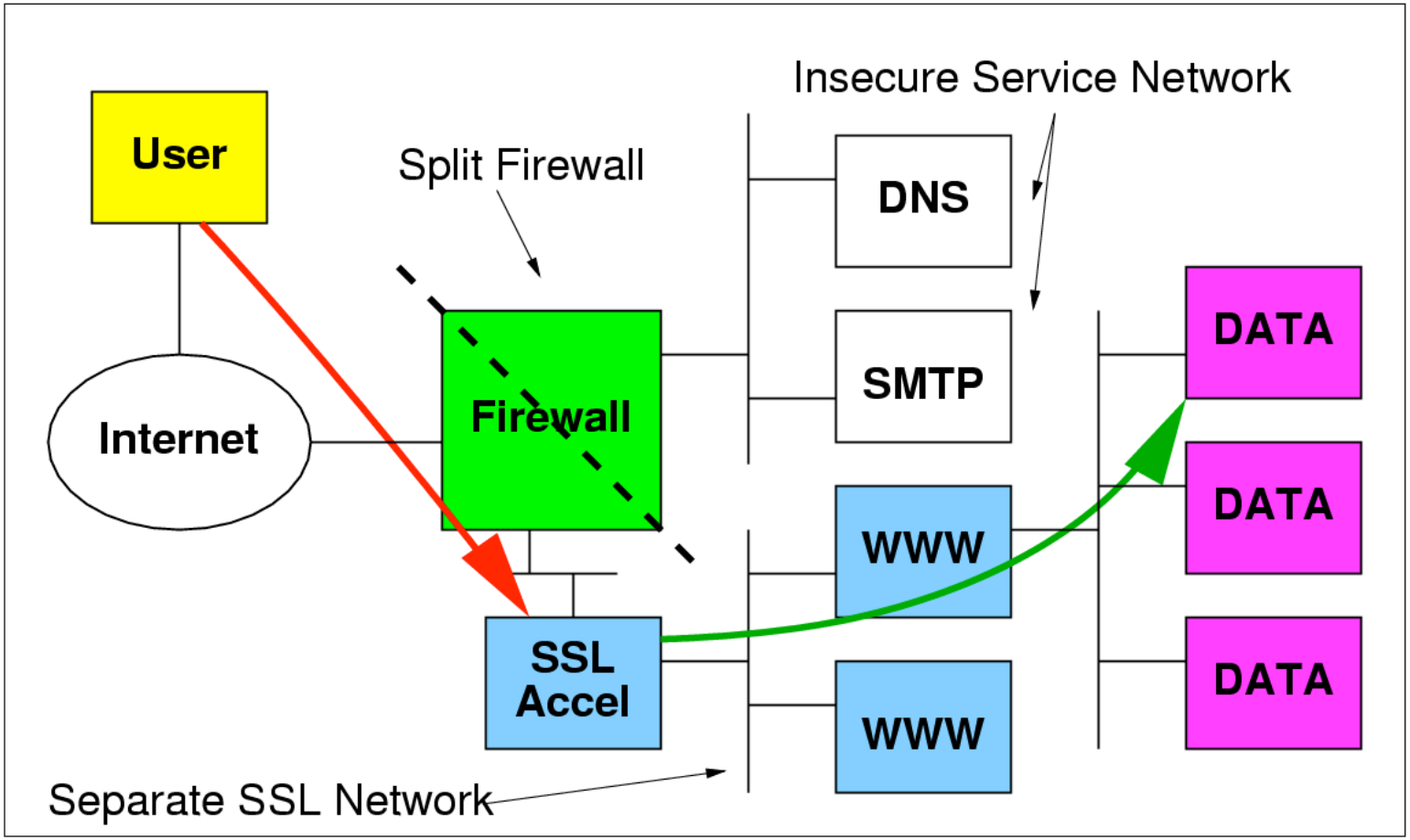
Design Example: Theory



Design Example: Oops!



Better Design



Solution?

- How do we address these challenges?
  - Clever Network Design
    - Bear risks in mind when laying-out architecture
    - Build so that (some) problems never arise
  - Clever Host Design
    - Build computers so they are less-subject to attack
    - Build computers for extra robustness
  - Overall: apply “Defence In Depth” philosophy
    - So, what is “Defence in Depth” ?

The Philosophy of “Defence In Depth”

Defence in Depth

- Motto #1
  - Use multiple, independent, different, mutually-reinforcing security technologies
- Motto #2
  - Use whatever works, is manageable and available, and configure them sensibly and as simply as possible
- Motto #3
  - Employ a “default-deny” approach
    - I.e., “you can only access that which we publish”

Defence in Depth

- Use of:
  - Multiple
  - Independent
  - Different
  - Mutually-Reinforcing
  - ...Security Technologies
- Not a 100% solution...
  - ...but nothing is!

Defence in Depth

- Compare: Castle Defences
  - Castle Video (6m33s)

Key Points

- Defence in Depth
  - Use of different technologies with different failure modes
  - Layers of security work to reduce profile available for attack

    You only see 10% of 10% of 10% of attacks …
  - There may be loss of some auditing information between layers, but…
  - “Are you doing security research, or are you trying to defend yourself?”

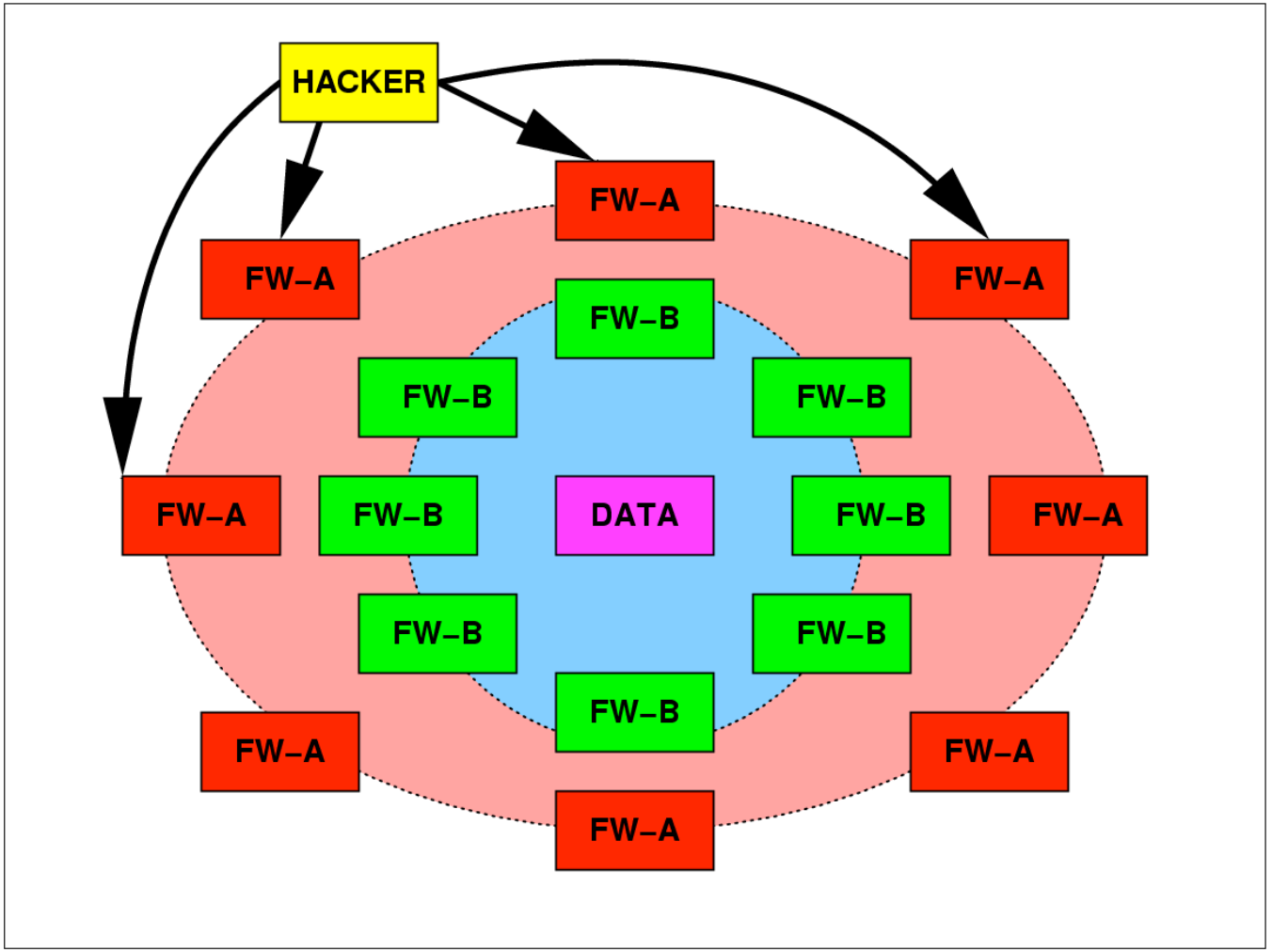
How Much Depth?

- When do I stop adding layers?
- Good question!
  - Depends upon what you are trying to protect.
- Judgement call
  - Personally, I reckon when all major risks have been mitigated twice, in different, independent ways, that's the minimum.

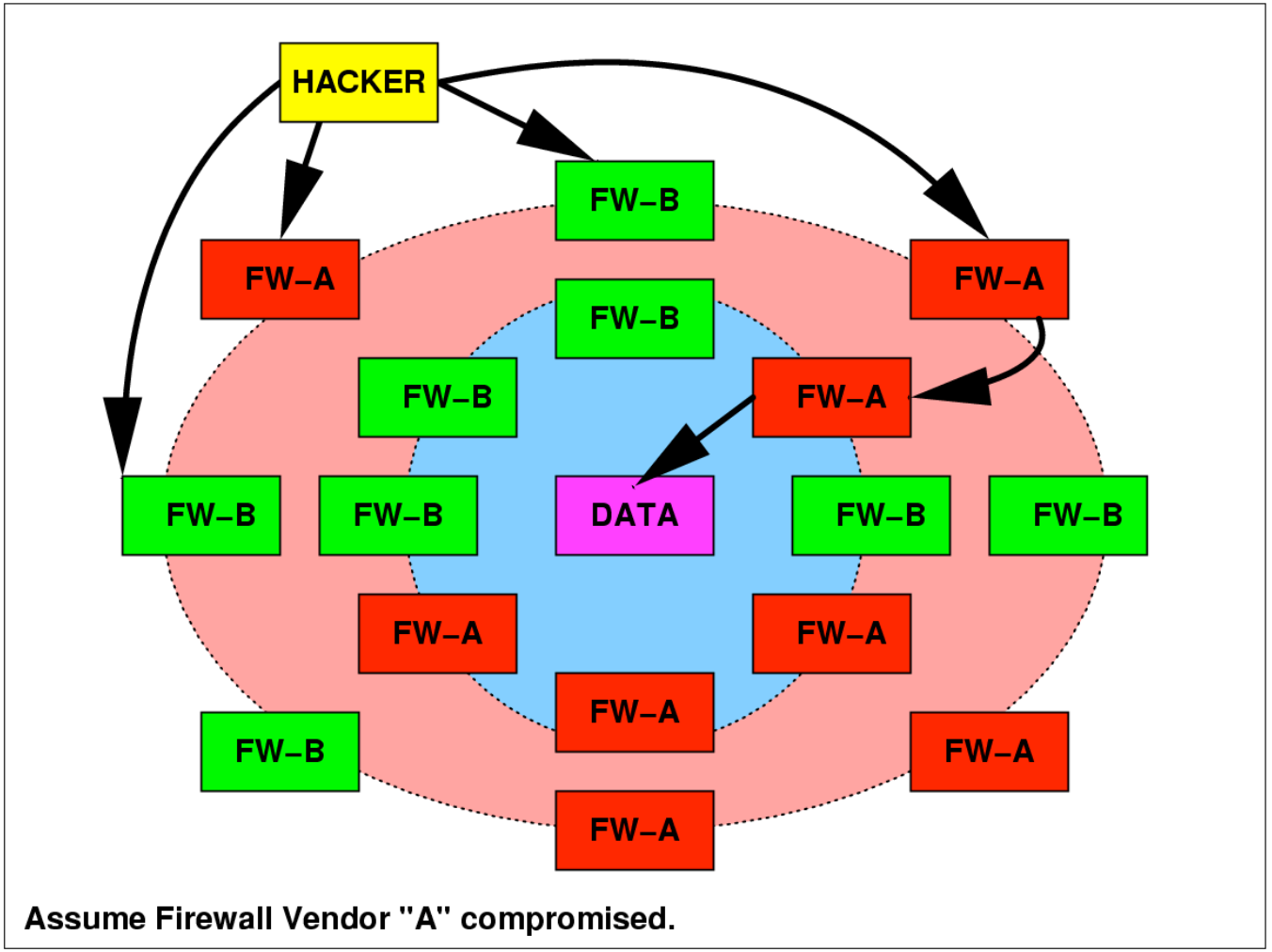
How robust is it?

- A very good approach to security...
- ...although it is not a 100% solution
  - ...its “ablative shield” approach yields better security than other “monolithic” solutions.
  - You will never get 100% security, anyway.
- But things can still go wrong...
  - For illustration...

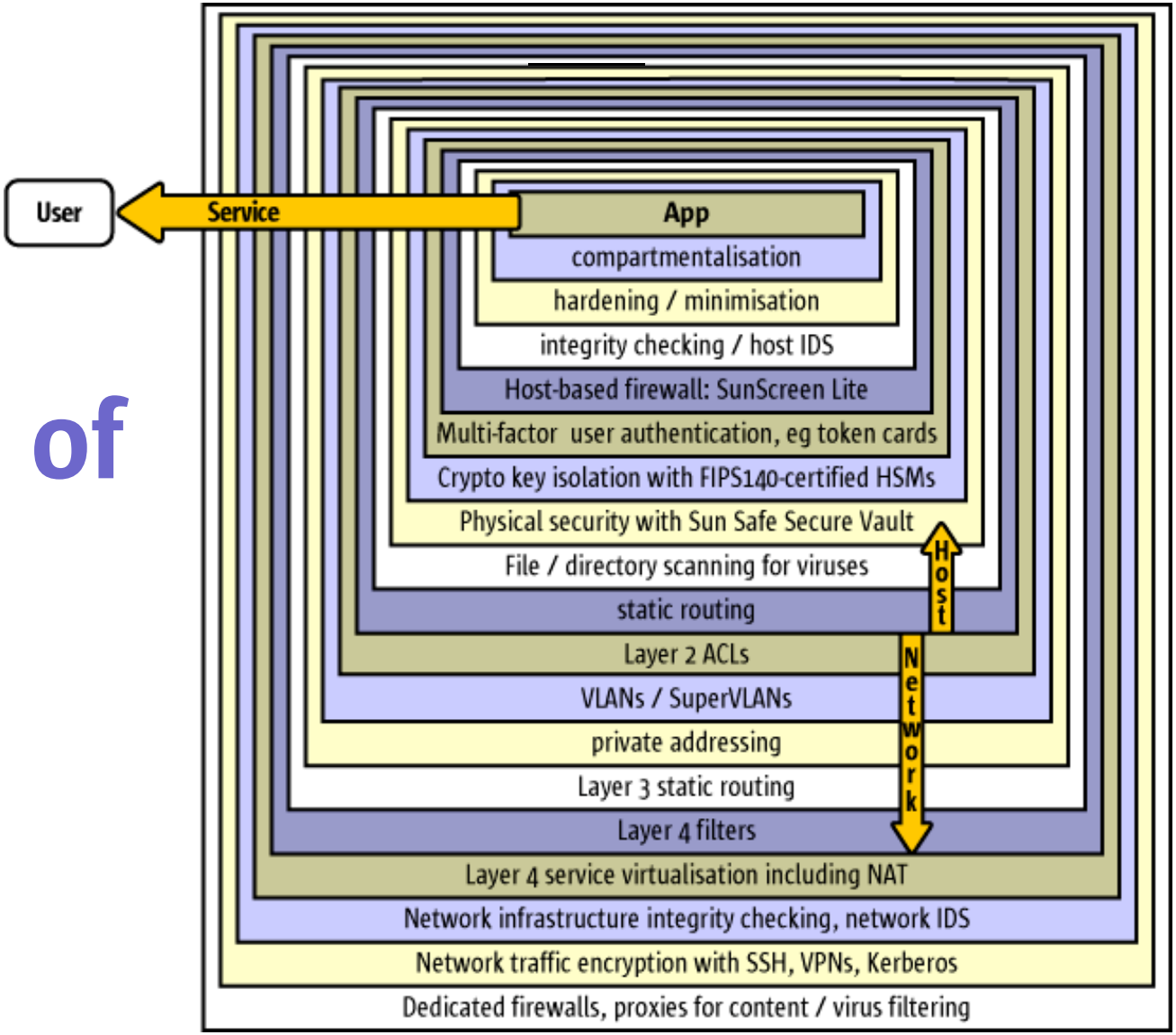
If Implemented Well...



If Implemented Badly...



Extremes of Available Choice



Summary

- Defence in Depth
  - Is a 7000-year-old approach to security that works really well
  - Avoids monolithic security issues and “monoculture syndrome”
  - Easy / inexpensive to build, but requires conscientious management and some forethought.
  - Investment in this methodology will last for a long time

Summary

- Security requires continual investment
  - Why audit, if you never read the logs?
  - Why have intrusion detection, if you don't want to wake up at 0300h?

    Together, these yield budget justification!
  - Why implement security, and yet fail to check its continued effectiveness over time?

    Healthchecks will yield ROI figures!
  - Why protect, if you do not value?

Truisms

- “Security is not a product...
  - ...it is a process!”

...or, personally speaking:

- “Security is not a process...
  - ...it is a lifestyle!”

Alec Muffett
Sun Professional Services
Alec.Muffett@Sun.COM

Sun microsystems
We make the net work.