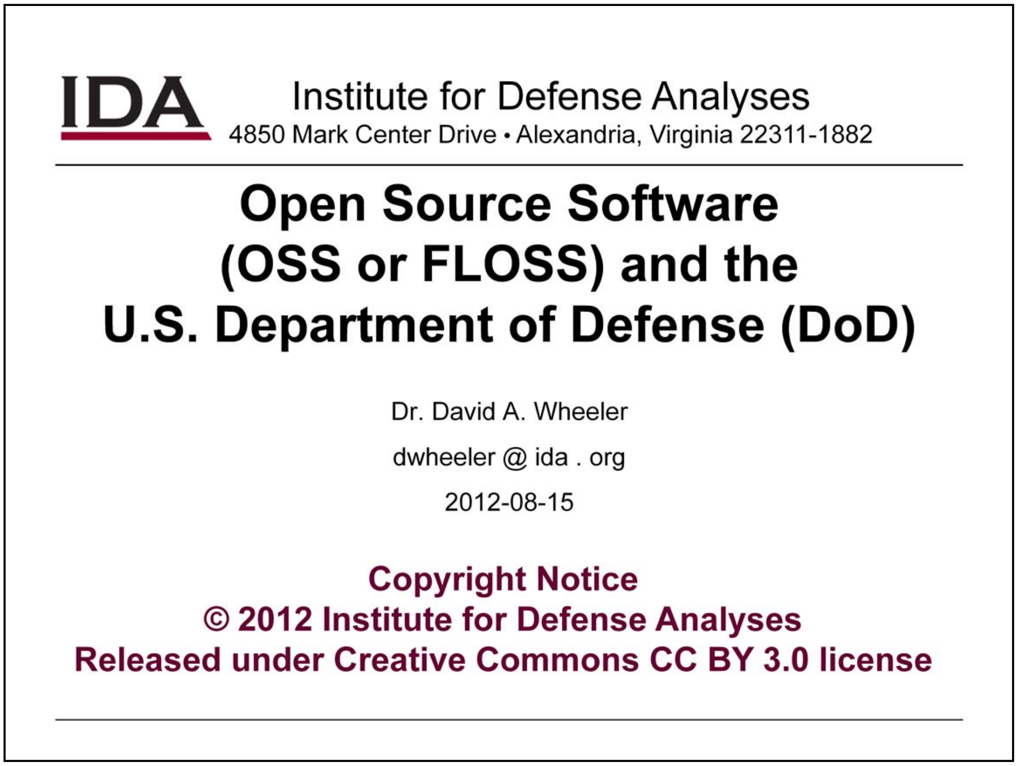


IDA Institute for Defense Analyses
4850 Mark Center Drive • Alexandria, Virginia 22311-1882

Open Source Software (OSS or FLOSS) and the U.S. Department of Defense (DoD)

Dr. David A. Wheeler

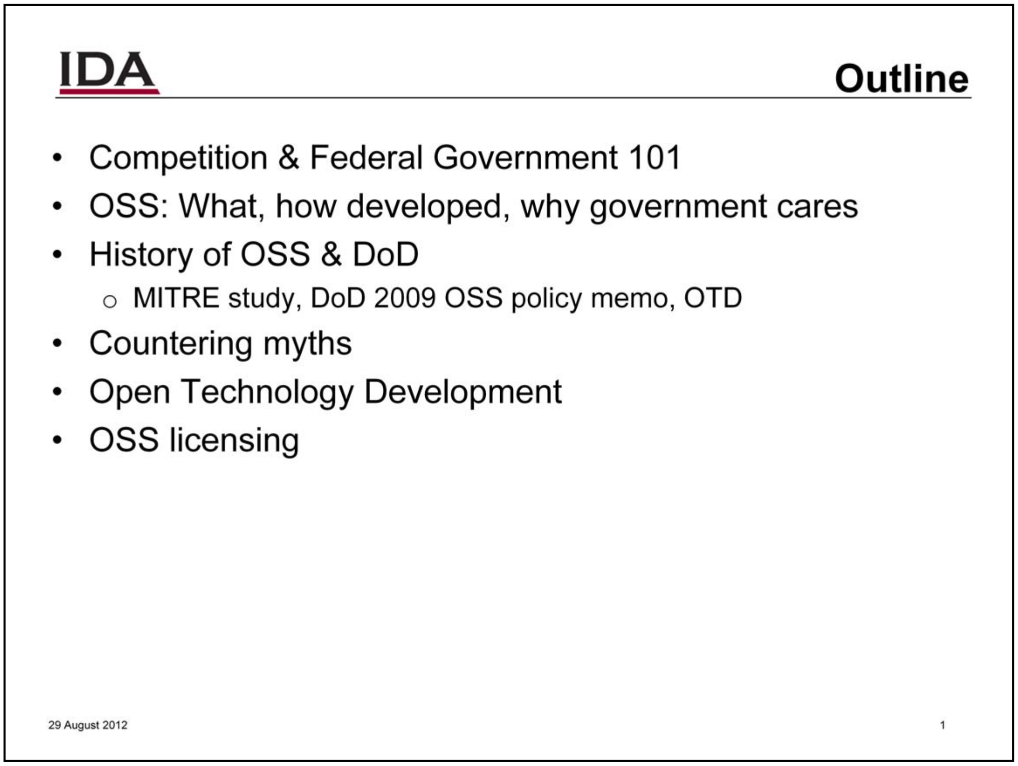
dwheeler @ ida . org

2012-08-15

This presentation was CLEARED for open publication on May 14, 2012, by the Office of Security Review, Department of Defense. Clearance was recommended by OUSD(AT&L)DIR ARA/EI, Bradrick Oeth, IDA Contracting Officers' Representative, of the office "ODD, Enterprise Information and OSD Studies" and agency OUSD(AT&L) DIR ARA/EI. OSR case 12-S-1973.

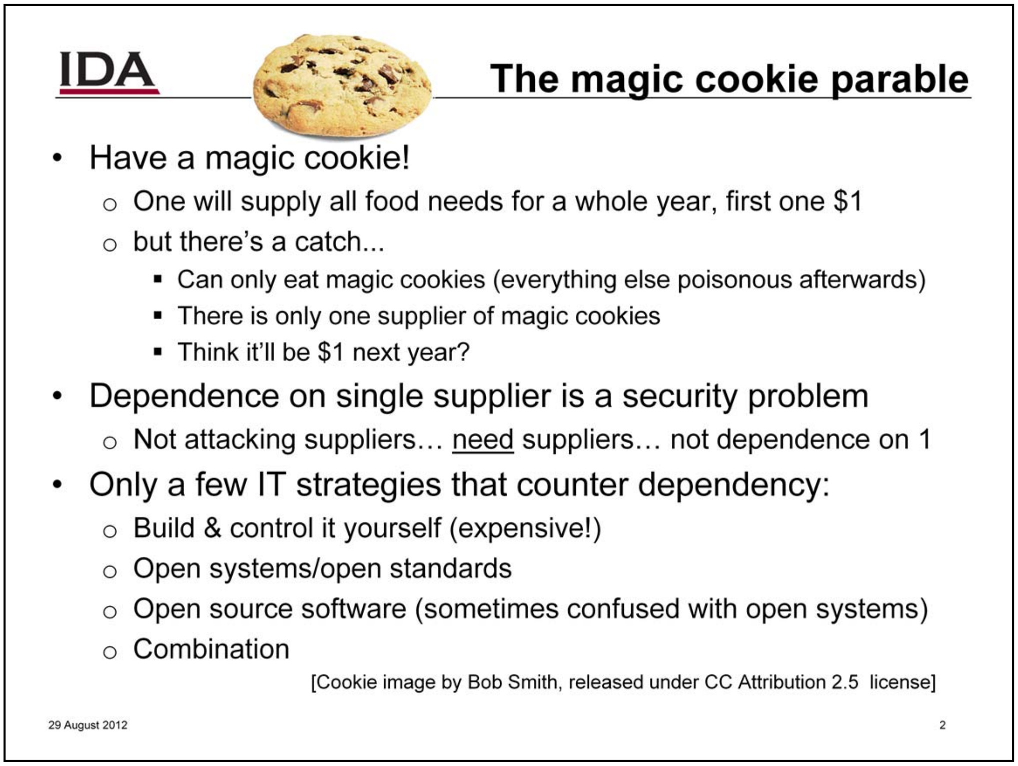
The email address of Dr. David A. Wheeler is dwheeler at ida dot org.

Text quotations and images are believed to meet established fair use criteria (17 U.S.C. §107), and are being used in "in a scholarly or technical work, for illustration or clarification of the author's observations. . . ." See United States Copyright Office, FL-102 (June 2012).



Outline

- Competition & Federal Government 101
- OSS: What, how developed, why government cares
- History of OSS & DoD
  - MITRE study, DoD 2009 OSS policy memo, OTD
- Countering myths
- Open Technology Development
- OSS licensing

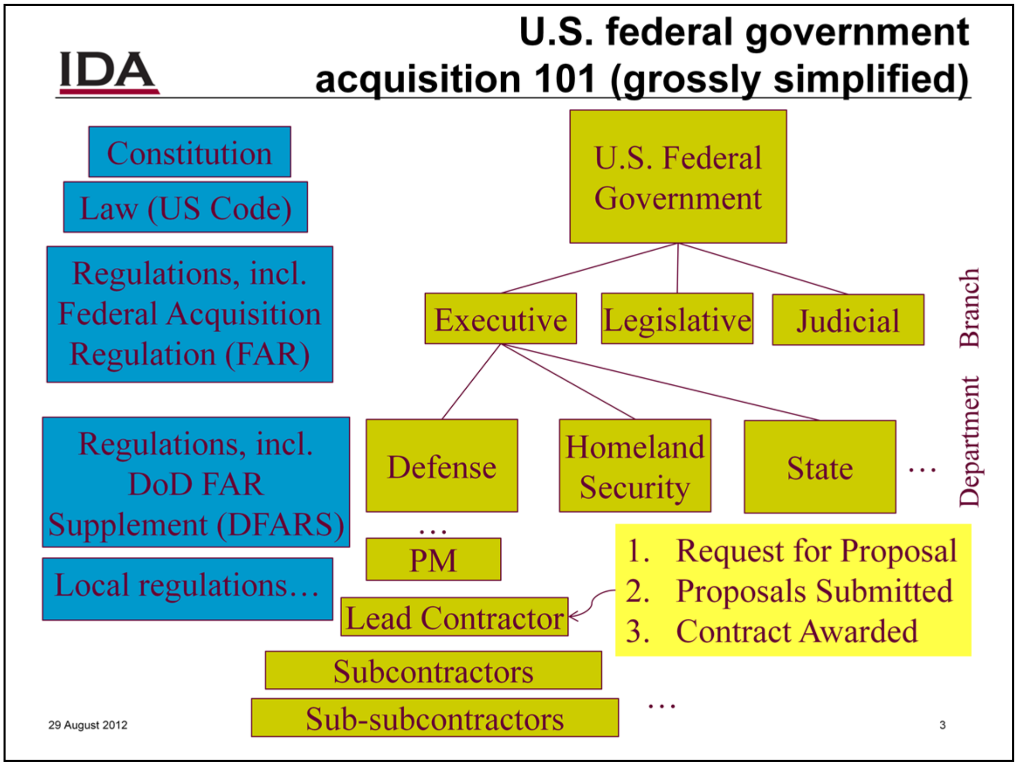


IDA

The magic cookie parable

- Have a magic cookie!
  - One will supply all food needs for a whole year, first one $1
  - but there’s a catch...
    - Can only eat magic cookies (everything else poisonous afterwards)
    - There is only one supplier of magic cookies
    - Think it’ll be $1 next year?
- Dependence on single supplier is a security problem
  - Not attacking suppliers… need suppliers… not dependence on 1
- Only a few IT strategies that counter dependency:
  - Build & control it yourself (expensive!)
  - Open systems/open standards
  - Open source software (sometimes confused with open systems)
  - Combination

[Cookie image by Bob Smith, released under CC Attribution 2.5 license]



IDA

U.S. federal government acquisition 101 (grossly simplified)

Constitution

Law (US Code)

Regulations, incl. Federal Acquisition Regulation (FAR)

Regulations, incl. DoD FAR Supplement (DFARS)

Local regulations…

U.S. Federal Government

Executive

Legislative

Judicial

Branch

Defense

Homeland Security

State

…

Department

…

PM

Lead Contractor

1. Request for Proposal
2. Proposals Submitted
3. Contract Awarded

Subcontractors

Sub-subcontractors

…

29 August 2012

3

The U.S. federal government is divided into three branches; the executive branch is divided into a number of departments and agencies. The Department of Defense (DoD) is further divided, typically down to program/project managers (PMs), who often have lead contractors, who often have subcontractors at one or more tiers.

Legally, these are governed at the top by the U.S. Constitution. Under this is the law (including the codified law – the US Code), and under this are various regulations such as the Federal Acquisition Regulation (FAR) that governs most acquisitions. There are often more and more local regulations, for example, the DoD has its DoD FAR Supplement (DFARS) that supplements the FAR.

Contractors are typically acquired through a 3-step process: The government sends out a request for proposal, proposals are submitted by various proposers, and then the government selects the “best” one and awards a contract. This is grossly oversimplified.

Federal IT dashboard is available at: http://www.itdashboard.gov/

(No hypertext link policy found.)

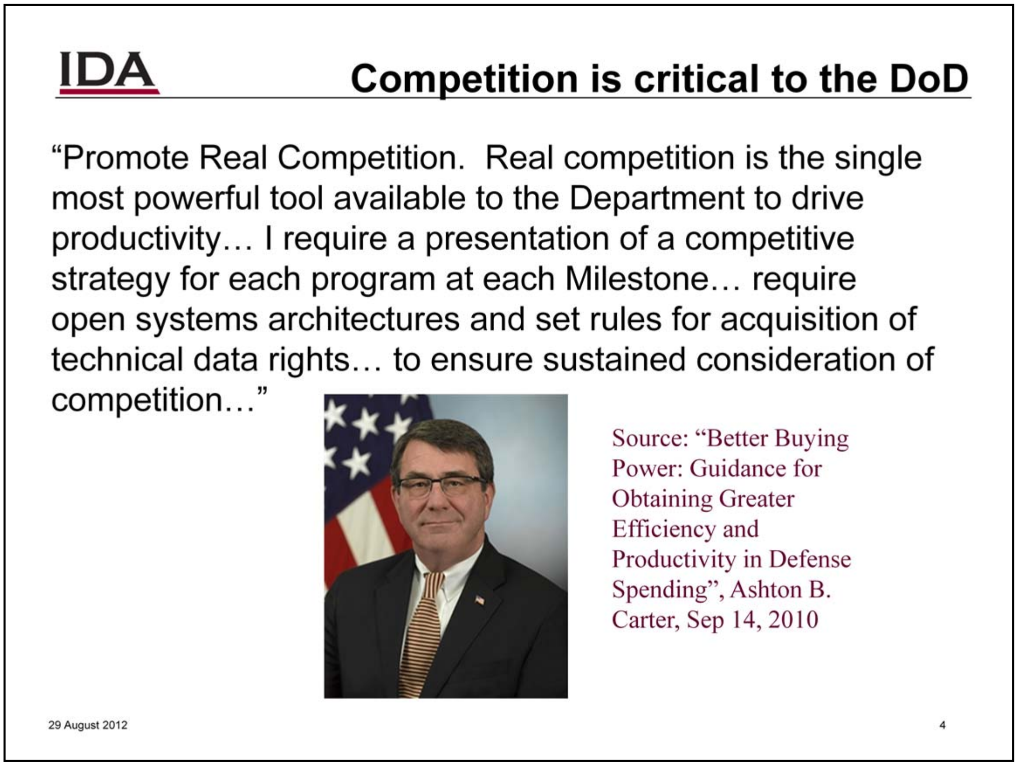
The records retrieved 2012-04-16 report that, among the IT major project investments of FY2012,

$75.0B was spent in the Executive branch; $33.8B in defense.

Gartner says Worldwide IT spending in 2011 $3661B (IT services $845B, Enterprise SW $267B), in 2012 $3751B

(IT services $856B, $472B) http://www.zdnet.com/blog/service-oriented/analysts-bullish-on-it-spending-where-will-the-money-go/8779

(No hypertext link policy found when searching for “link” in their FAQ.)



Competition is critical to the DoD

"Promote Real Competition. Real competition is the single most powerful tool available to the Department to drive productivity… I require a presentation of a competitive strategy for each program at each Milestone… require open systems architectures and set rules for acquisition of technical data rights… to ensure sustained consideration of competition…"

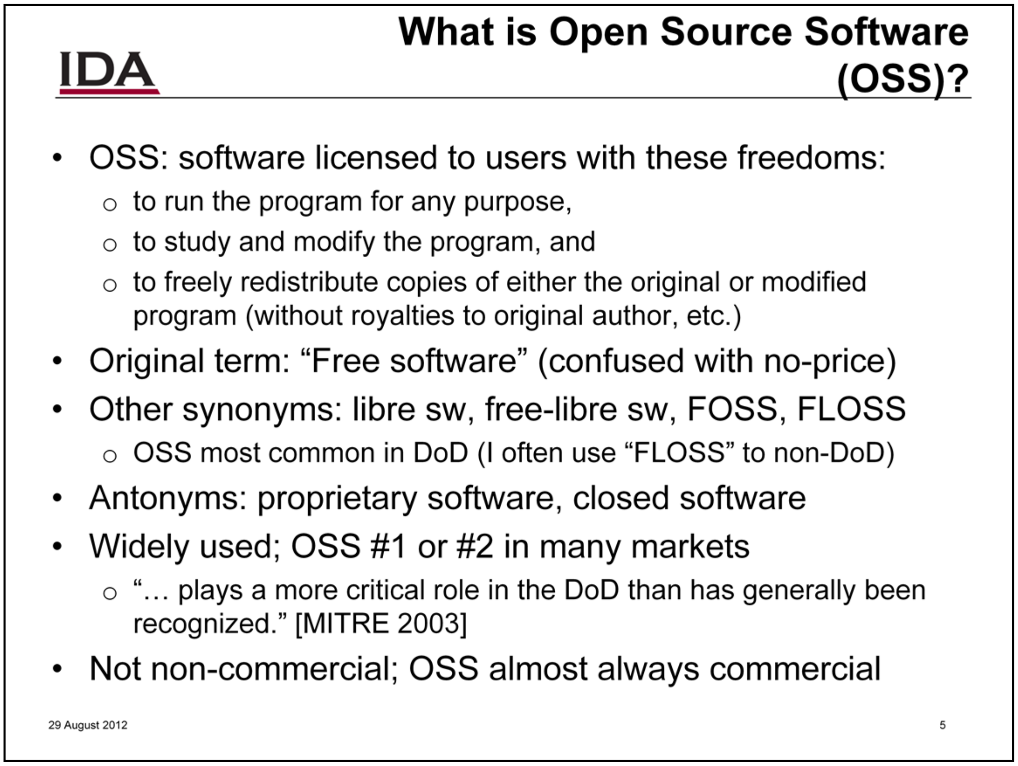
Source: "Better Buying Power: Guidance for Obtaining Greater Efficiency and Productivity in Defense Spending", Ashton B. Carter, Sep 14, 2010

29 August 2012

4

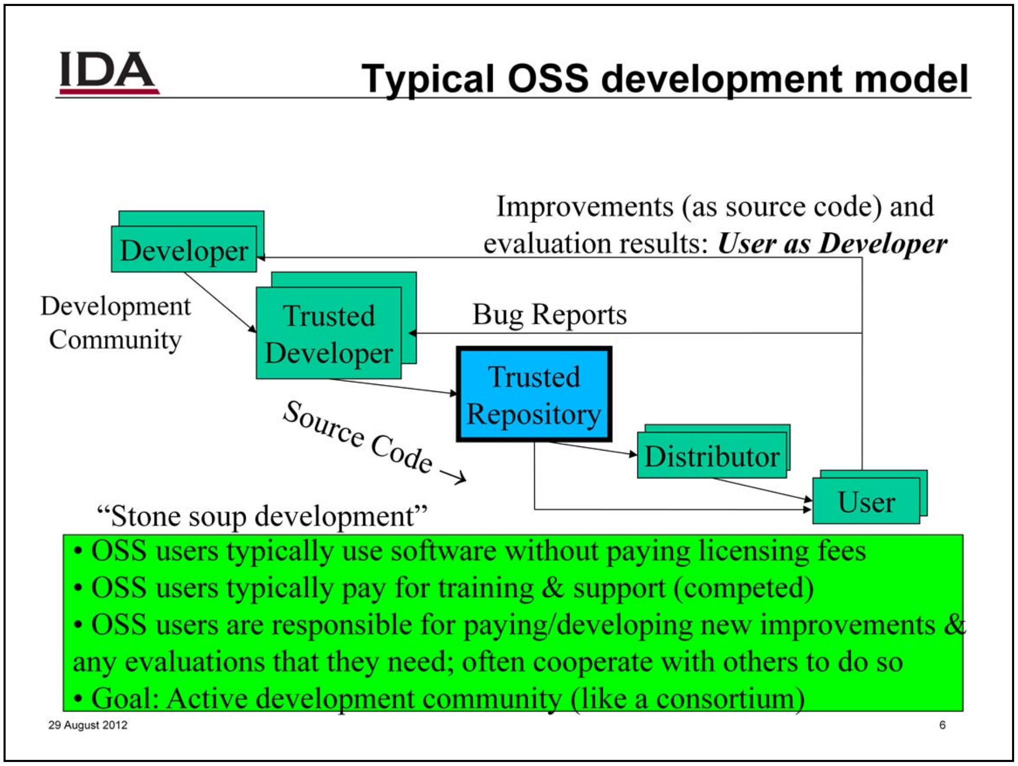
Image from:
http://www.defense.gov/bios/biographydetail.aspx?biographyid=186

No special terms of use were found, including for hyperlinks.



What is Open Source Software (OSS)?

- OSS: software licensed to users with these freedoms:
  - to run the program for any purpose,
  - to study and modify the program, and
  - to freely redistribute copies of either the original or modified program (without royalties to original author, etc.)
- Original term: “Free software” (confused with no-price)
- Other synonyms: libre sw, free-libre sw, FOSS, FLOSS
  - OSS most common in DoD (I often use “FLOSS” to non-DoD)
- Antonyms: proprietary software, closed software
- Widely used; OSS #1 or #2 in many markets
  - “… plays a more critical role in the DoD than has generally been recognized.” [MITRE 2003]
- Not non-commercial; OSS almost always commercial



IDA

Typical OSS development model

Improvements (as source code) and evaluation results: ***User as Developer***

Developer

Development Community

Trusted Developer

Bug Reports

Trusted Repository

Source Code →

Distributor

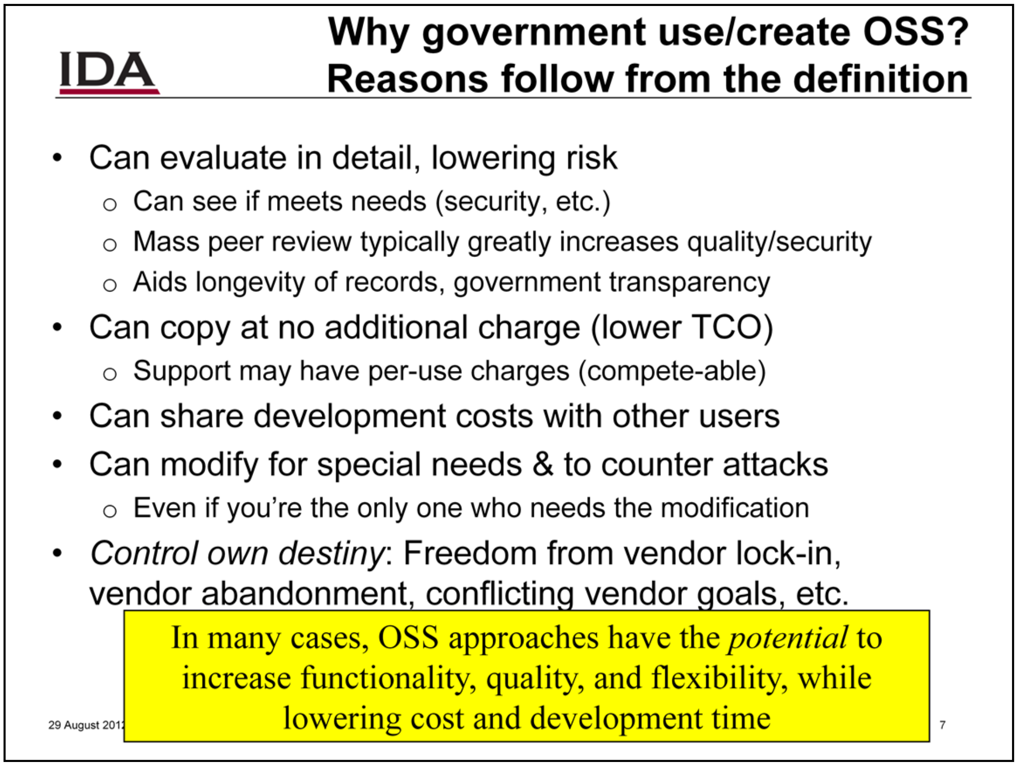
User

“Stone soup development”

- OSS users typically use software without paying licensing fees
- OSS users typically pay for training & support (competed)
- OSS users are responsible for paying/developing new improvements & any evaluations that they need; often cooperate with others to do so
- Goal: Active development community (like a consortium)

29 August 2012

6

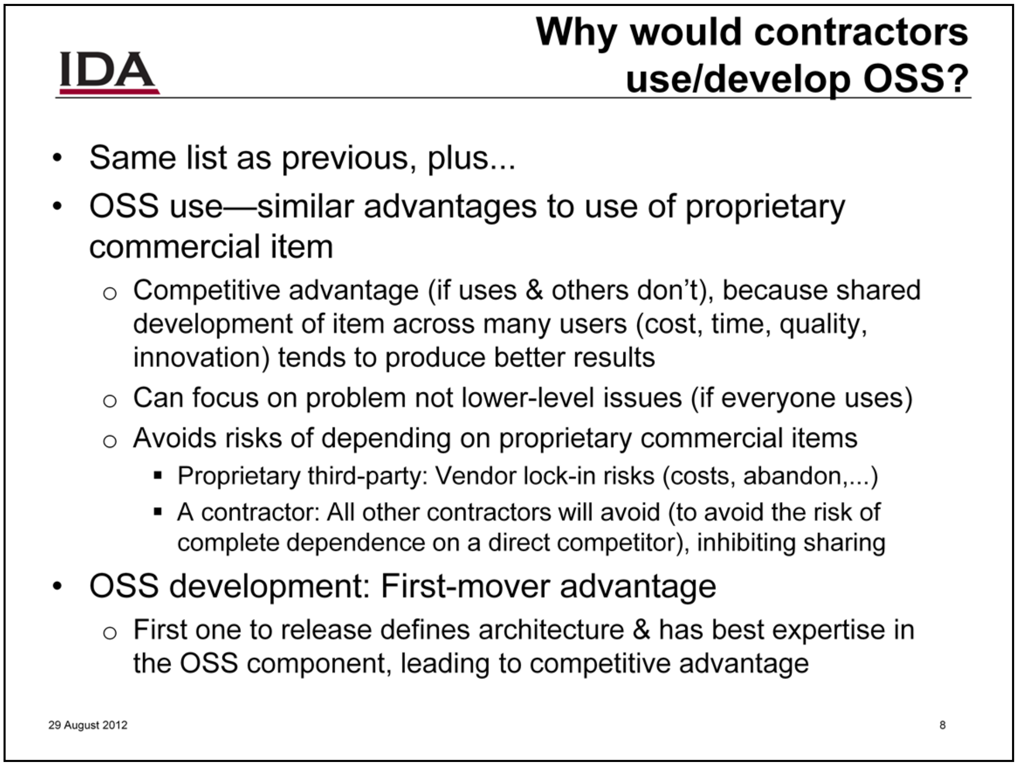


IDA

Why government use/create OSS? Reasons follow from the definition

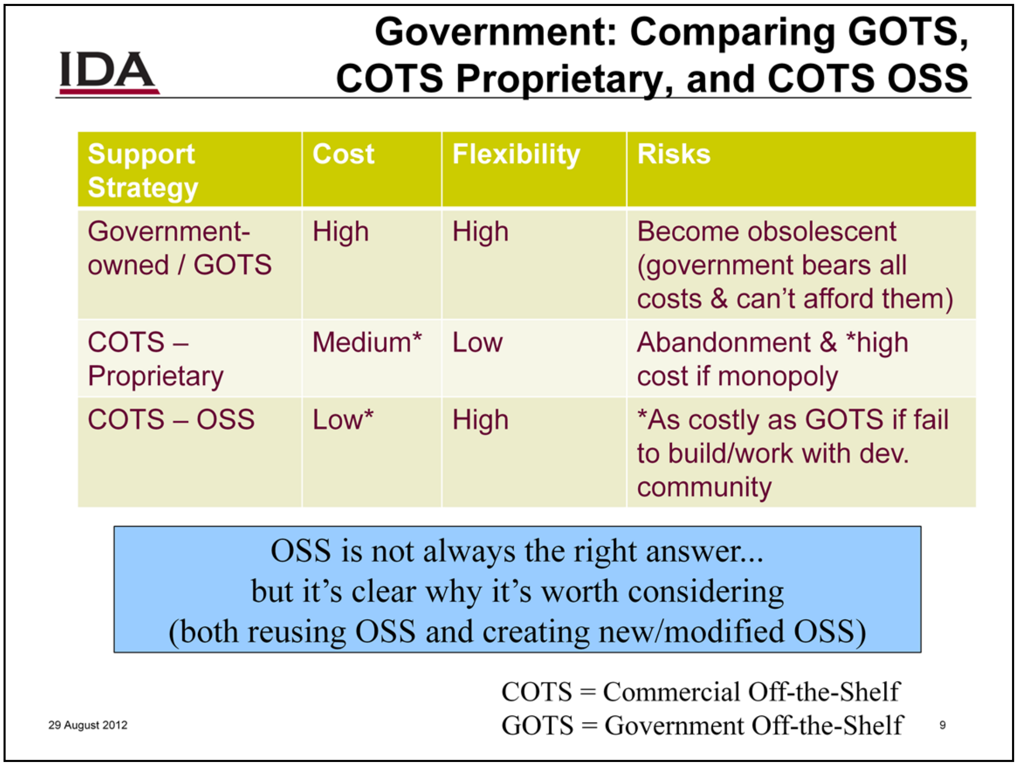
- Can evaluate in detail, lowering risk
  - Can see if meets needs (security, etc.)
  - Mass peer review typically greatly increases quality/security
  - Aids longevity of records, government transparency
- Can copy at no additional charge (lower TCO)
  - Support may have per-use charges (compete-able)
- Can share development costs with other users
- Can modify for special needs & to counter attacks
  - Even if you're the only one who needs the modification
- *Control own destiny*: Freedom from vendor lock-in, vendor abandonment, conflicting vendor goals, etc.

In many cases, OSS approaches have the *potential* to increase functionality, quality, and flexibility, while lowering cost and development time



Why would contractors use/develop OSS?

- Same list as previous, plus...
- OSS use—similar advantages to use of proprietary commercial item
  - Competitive advantage (if uses & others don’t), because shared development of item across many users (cost, time, quality, innovation) tends to produce better results
  - Can focus on problem not lower-level issues (if everyone uses)
  - Avoids risks of depending on proprietary commercial items
    - Proprietary third-party: Vendor lock-in risks (costs, abandon,...)
    - A contractor: All other contractors will avoid (to avoid the risk of complete dependence on a direct competitor), inhibiting sharing
- OSS development: First-mover advantage
  - First one to release defines architecture & has best expertise in the OSS component, leading to competitive advantage



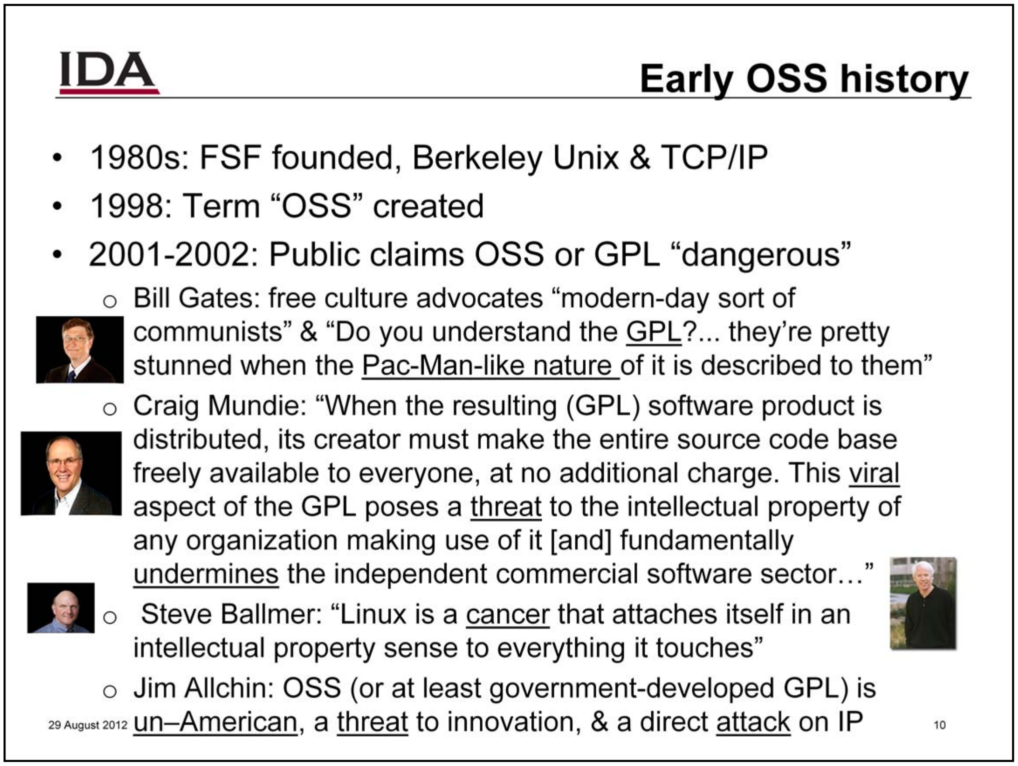
IDA

Government: Comparing GOTS, COTS Proprietary, and COTS OSS

Support Strategy	Cost	Flexibility	Risks
Government-owned / GOTS	High	High	Become obsolescent (government bears all costs & can't afford them)
COTS – Proprietary	Medium*	Low	Abandonment & *high cost if monopoly
COTS – OSS	Low*	High	*As costly as GOTS if fail to build/work with dev. community

OSS is not always the right answer...
but it's clear why it's worth considering
(both reusing OSS and creating new/modified OSS)

COTS = Commercial Off-the-Shelf
GOTS = Government Off-the-Shelf



IDA

Early OSS history

- 1980s: FSF founded, Berkeley Unix & TCP/IP
- 1998: Term “OSS” created
- 2001-2002: Public claims OSS or GPL “dangerous”
  - Bill Gates: free culture advocates “modern-day sort of communists” & “Do you understand the GPL?... they’re pretty stunned when the Pac-Man-like nature of it is described to them”
  - Craig Mundie: “When the resulting (GPL) software product is distributed, its creator must make the entire source code base freely available to everyone, at no additional charge. This viral aspect of the GPL poses a threat to the intellectual property of any organization making use of it [and] fundamentally undermines the independent commercial software sector…”
  - Steve Ballmer: “Linux is a cancer that attaches itself in an intellectual property sense to everything it touches”
  - Jim Allchin: OSS (or at least government-developed GPL) is un–American, a threat to innovation, & a direct attack on IP

29 August 2012

10

Quotations are from the following sources:

“Modern-day sort of communists” quote from “Bill Gates: Free Culture advocates = Commies” by Xeni Jardin, Wednesday, Jan 5th at 8:30pm, http://boingboing.net/2005/01/05/bill-gates-free-cult.html

Boing Boing’s general policy (http://boingboing.net/policies) links to its “policy” for hypertext linking, which mocks the idea of having a policy on hypertext linking (e.g., “Boing Boing doesn't believe in linking policies. They're dangerous, have no basis in law, and they break the norms that make the Web possible”) (http://boingboing.net/2004/10/04/boing-boing-has-a-li.html).

“GPL Pacman will eat your business, warns Gates: Timely warning from The Great Engulfer himself” by John Lettice , 20th June 2001 14:06 GMT, http://www.theregister.co.uk/2001/06/20/gpl_pacman_will_eat_your/

The terms and conditions of “http://www.theregister.co.uk/2008/09/30/reg_ts_and_cs/” do not limit the use of hypertext links.

Craig Mundie’s speech, plus commentary, is available here:

http://mindplusplus.wordpress.com/2002/10/26/my_response_to__1/

No hypertext linking policy is noted there.

“Ballmer: ‘Linux is a cancer’: Contaminates all other software with Hippie GPL rubbish” by Thomas C Greene, 2nd June 2001 18:19 GMT,

http://www.theregister.co.uk/2001/06/02/ballmer_linux_is_a_cancer/

Clarifications on Jim Allchin’s comments are in:

“Microsoft Defends Jim Allchin's Comments” by Seumas, Feb 21, 2001 at 11:23:39 AM EST

http://www.kuro5hin.org/?op=displaystory;sid=2001/2/21/41423/3184

There is no evidence of a hypertext linking policy at: http://www.kuro5hin.org/special/faq

“Microsoft, co-author of the Linux kernel”

http://www.dwheeler.com/blog/2011/07/14/#microsoft-linux-author

The hypertext linking policy says “You don't need to ask permission to link to the material on my website; *just link to it*.” (http://www.dwheeler.com/aboutsite.html)

Images of Bill Gates, Craig Mundie, and Jim Allchin courtesy of Microsoft’s press material:

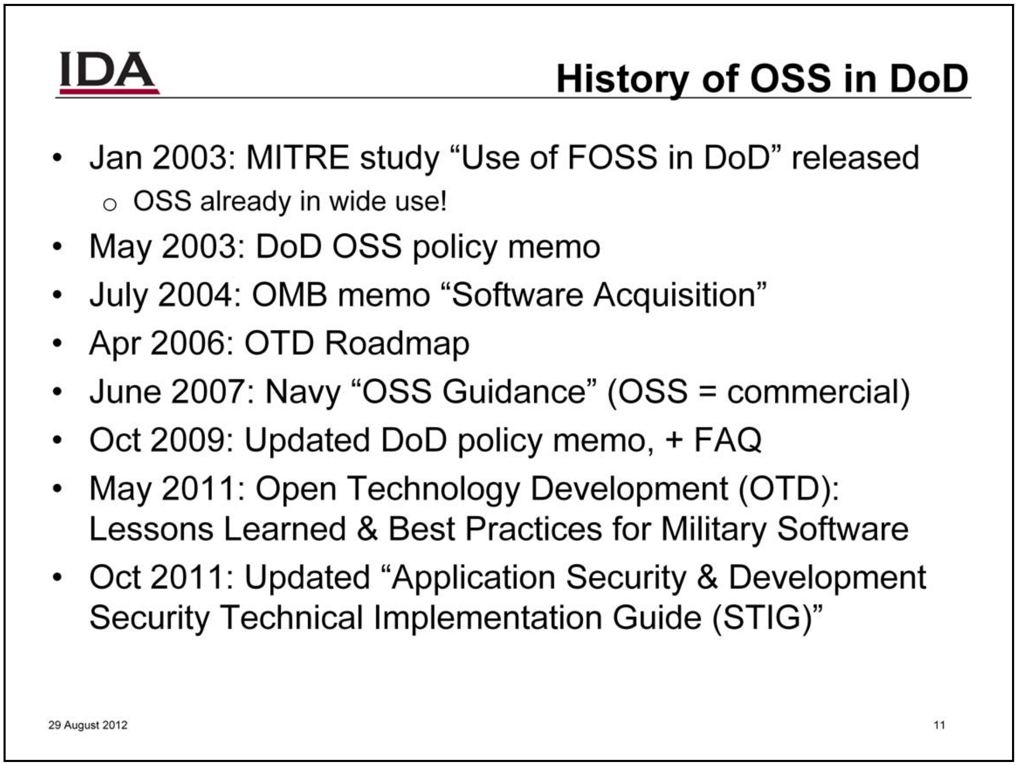
http://www.microsoft.com/presspass/exec/craig/

http://www.microsoft.com/presspass/exec/jim/default.mspx

http://www.microsoft.com/presspass/exec/billg/

The website terms of use do not say anything about quoting hypertext links (http://www.microsoft.com/About/Legal/EN/US/IntellectualProperty/Copyright/default.aspx).

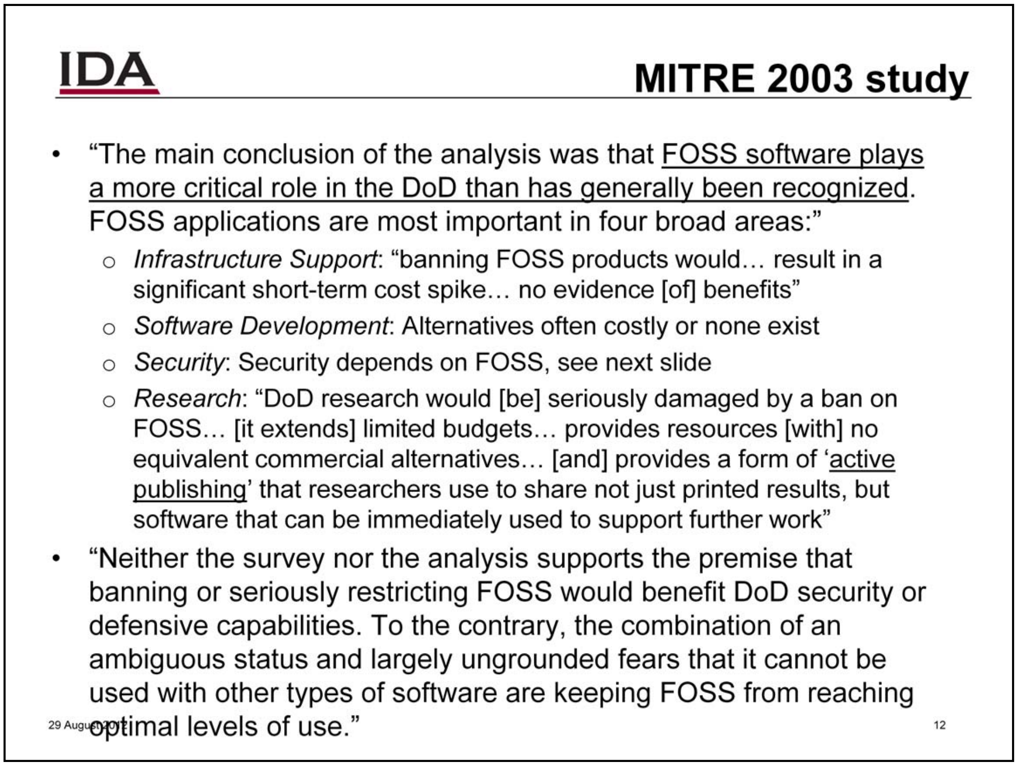
The list of quotes given here, and the underlining, were previously reported in the IDA presentation “Why the GPL *might* not Destroy the Universe”.



IDA

History of OSS in DoD

- Jan 2003: MITRE study “Use of FOSS in DoD” released
  - OSS already in wide use!
- May 2003: DoD OSS policy memo
- July 2004: OMB memo “Software Acquisition”
- Apr 2006: OTD Roadmap
- June 2007: Navy “OSS Guidance” (OSS = commercial)
- Oct 2009: Updated DoD policy memo, + FAQ
- May 2011: Open Technology Development (OTD): Lessons Learned & Best Practices for Military Software
- Oct 2011: Updated “Application Security & Development Security Technical Implementation Guide (STIG)”

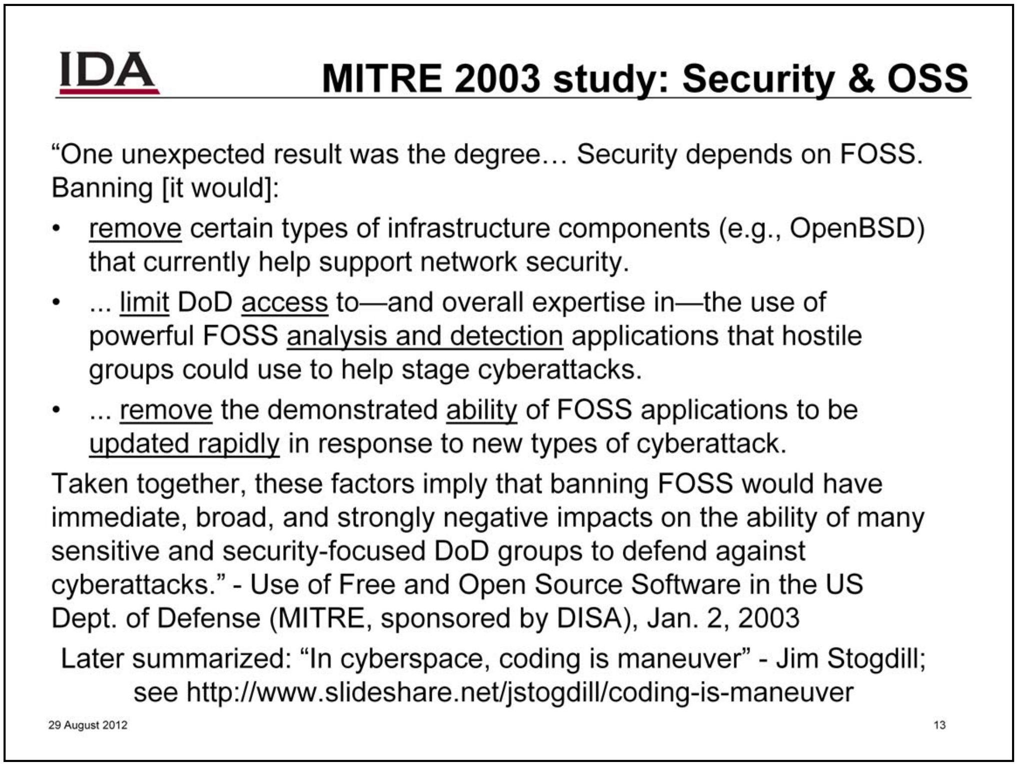


IDA

MITRE 2003 study

- "The main conclusion of the analysis was that FOSS software plays a more critical role in the DoD than has generally been recognized. FOSS applications are most important in four broad areas:"
  - *Infrastructure Support*: "banning FOSS products would… result in a significant short-term cost spike… no evidence [of] benefits"
  - *Software Development*: Alternatives often costly or none exist
  - *Security*: Security depends on FOSS, see next slide
  - *Research*: "DoD research would [be] seriously damaged by a ban on FOSS… [it extends] limited budgets… provides resources [with] no equivalent commercial alternatives… [and] provides a form of 'active publishing' that researchers use to share not just printed results, but software that can be immediately used to support further work"
- "Neither the survey nor the analysis supports the premise that banning or seriously restricting FOSS would benefit DoD security or defensive capabilities. To the contrary, the combination of an ambiguous status and largely ungrounded fears that it cannot be used with other types of software are keeping FOSS from reaching optimal levels of use."

All quotes here from the MITRE study "Use of FOSS in DoD"



MITRE 2003 study: Security & OSS

“One unexpected result was the degree… Security depends on FOSS. Banning [it would]:

- remove certain types of infrastructure components (e.g., OpenBSD) that currently help support network security.
- ... limit DoD access to—and overall expertise in—the use of powerful FOSS analysis and detection applications that hostile groups could use to help stage cyberattacks.
- ... remove the demonstrated ability of FOSS applications to be updated rapidly in response to new types of cyberattack.

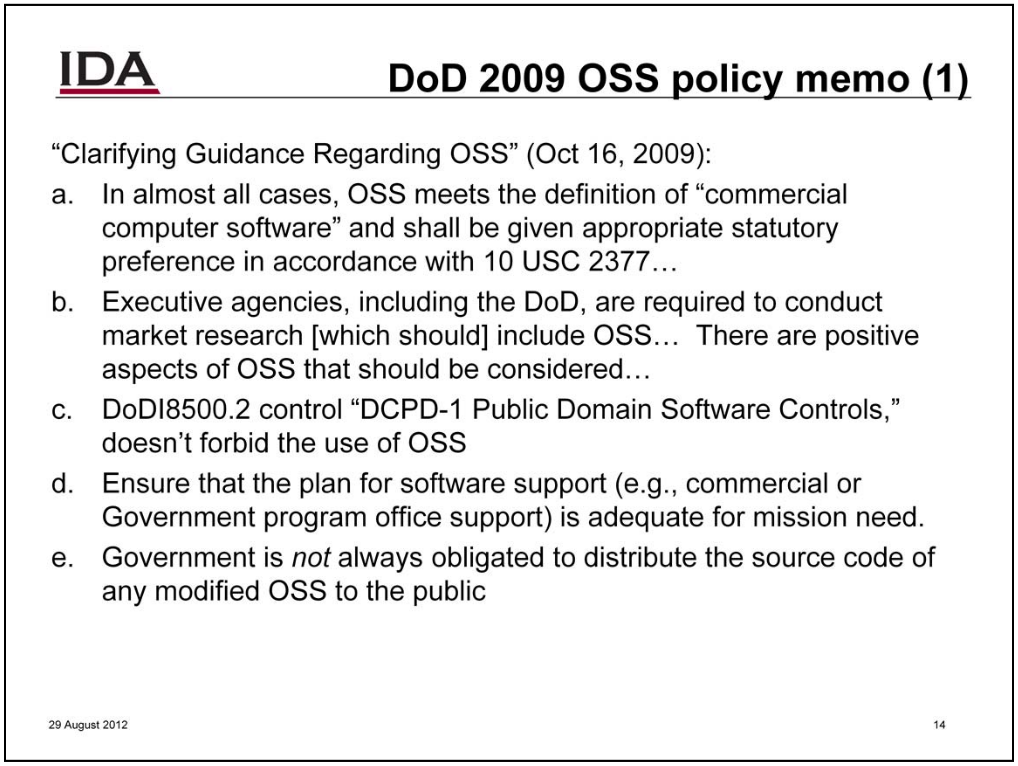
Taken together, these factors imply that banning FOSS would have immediate, broad, and strongly negative impacts on the ability of many sensitive and security-focused DoD groups to defend against cyberattacks.” - Use of Free and Open Source Software in the US Dept. of Defense (MITRE, sponsored by DISA), Jan. 2, 2003

Later summarized: “In cyberspace, coding is maneuver” - Jim Stogdill; see http://www.slideshare.net/jstogdill/coding-is-maneuver

29 August 2012 13

All quotes here from the MITRE study “Use of FOSS in DoD”.

The slideshare.net site terms of use do not say anything about quoting hypertext links (http://www.slideshare.net/terms).



DoD 2009 OSS policy memo (1)

“Clarifying Guidance Regarding OSS” (Oct 16, 2009):

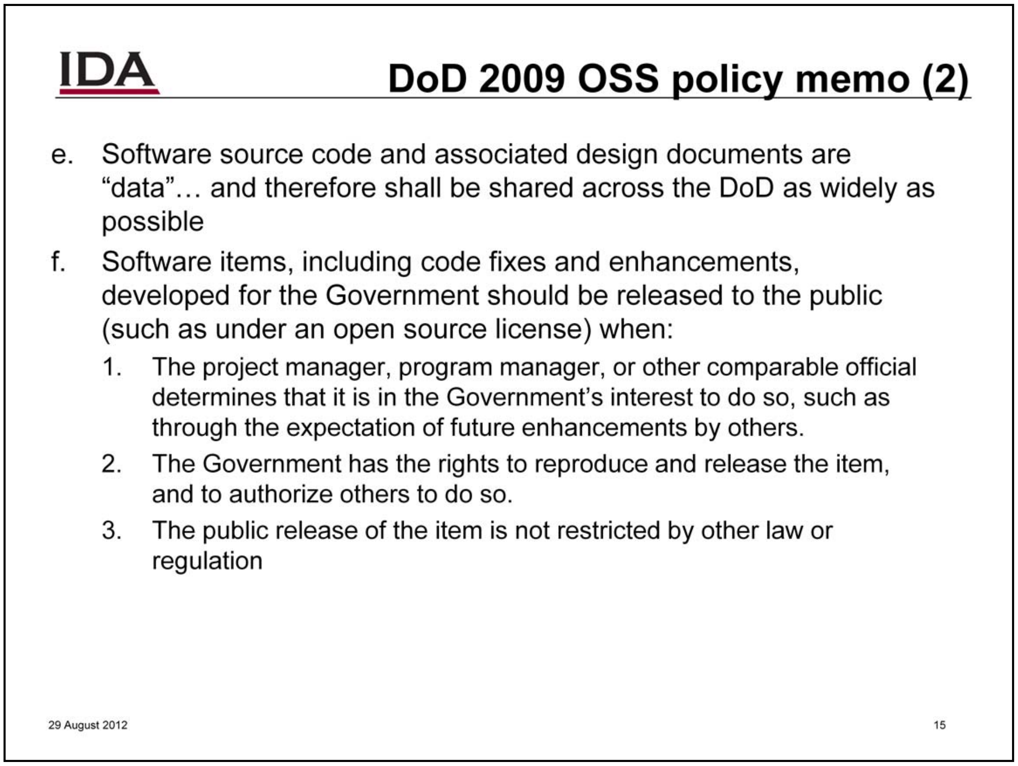
a. In almost all cases, OSS meets the definition of “commercial computer software” and shall be given appropriate statutory preference in accordance with 10 USC 2377…

b. Executive agencies, including the DoD, are required to conduct market research [which should] include OSS… There are positive aspects of OSS that should be considered…

c. DoDI8500.2 control “DCPD-1 Public Domain Software Controls,” doesn’t forbid the use of OSS

d. Ensure that the plan for software support (e.g., commercial or Government program office support) is adequate for mission need.

e. Government is *not* always obligated to distribute the source code of any modified OSS to the public



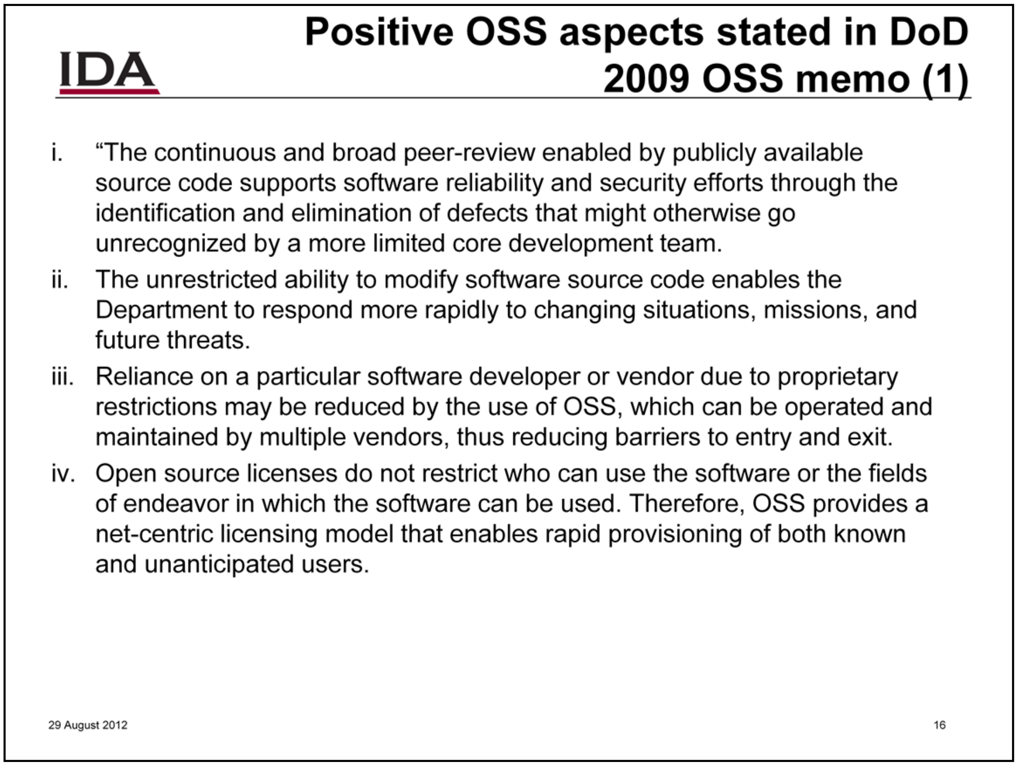
IDA

DoD 2009 OSS policy memo (2)

e. Software source code and associated design documents are “data”… and therefore shall be shared across the DoD as widely as possible

f. Software items, including code fixes and enhancements, developed for the Government should be released to the public (such as under an open source license) when:

   1. The project manager, program manager, or other comparable official determines that it is in the Government’s interest to do so, such as through the expectation of future enhancements by others.
   2. The Government has the rights to reproduce and release the item, and to authorize others to do so.
   3. The public release of the item is not restricted by other law or regulation



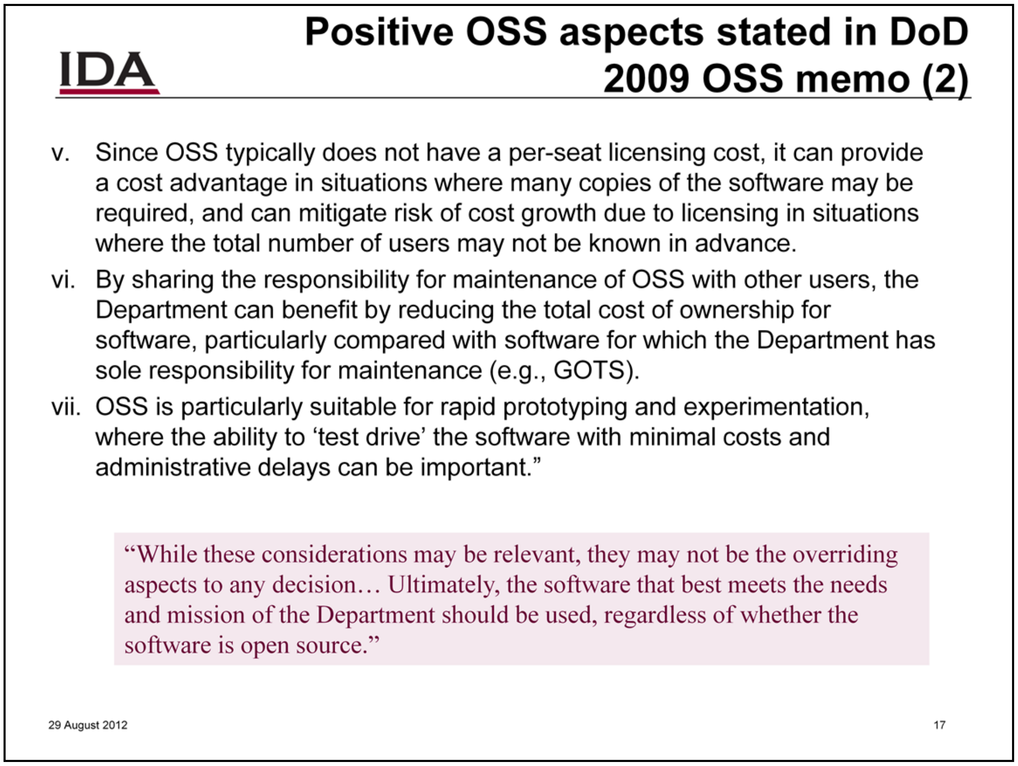
Positive OSS aspects stated in DoD 2009 OSS memo (1)

i. “The continuous and broad peer-review enabled by publicly available source code supports software reliability and security efforts through the identification and elimination of defects that might otherwise go unrecognized by a more limited core development team.

ii. The unrestricted ability to modify software source code enables the Department to respond more rapidly to changing situations, missions, and future threats.

iii. Reliance on a particular software developer or vendor due to proprietary restrictions may be reduced by the use of OSS, which can be operated and maintained by multiple vendors, thus reducing barriers to entry and exit.

iv. Open source licenses do not restrict who can use the software or the fields of endeavor in which the software can be used. Therefore, OSS provides a net-centric licensing model that enables rapid provisioning of both known and unanticipated users.



Positive OSS aspects stated in DoD 2009 OSS memo (2)

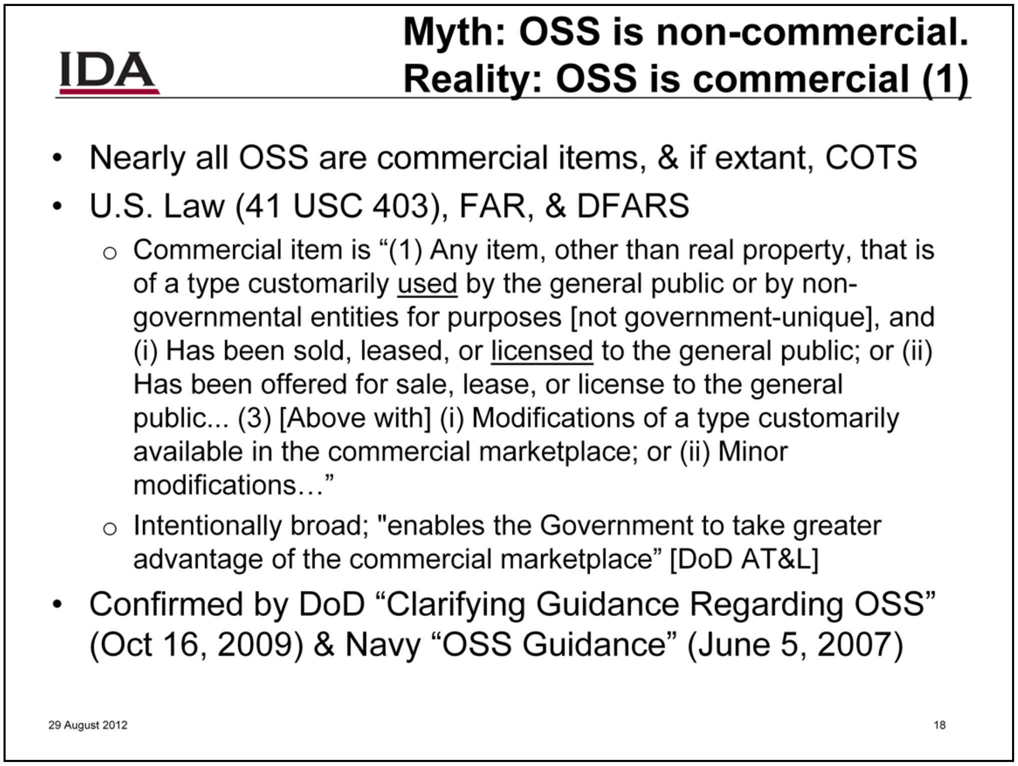
v. Since OSS typically does not have a per-seat licensing cost, it can provide a cost advantage in situations where many copies of the software may be required, and can mitigate risk of cost growth due to licensing in situations where the total number of users may not be known in advance.

vi. By sharing the responsibility for maintenance of OSS with other users, the Department can benefit by reducing the total cost of ownership for software, particularly compared with software for which the Department has sole responsibility for maintenance (e.g., GOTS).

vii. OSS is particularly suitable for rapid prototyping and experimentation, where the ability to 'test drive' the software with minimal costs and administrative delays can be important."

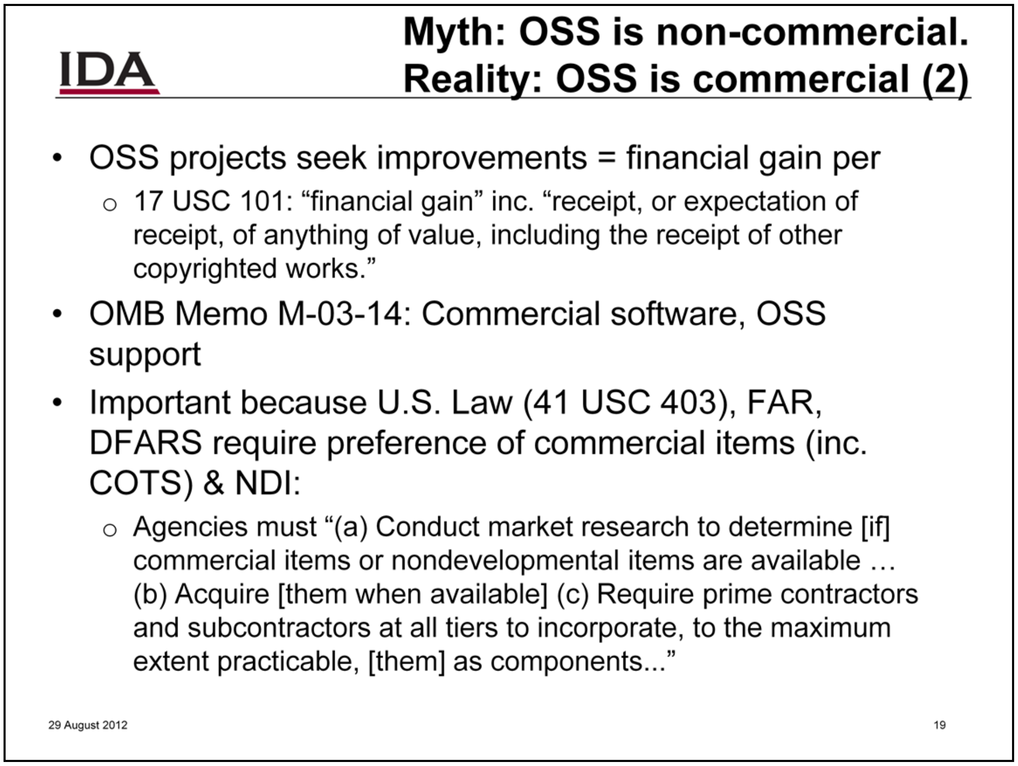
> "While these considerations may be relevant, they may not be the overriding aspects to any decision… Ultimately, the software that best meets the needs and mission of the Department should be used, regardless of whether the software is open source."

29 August 2012 17



Myth: OSS is non-commercial. Reality: OSS is commercial (1)

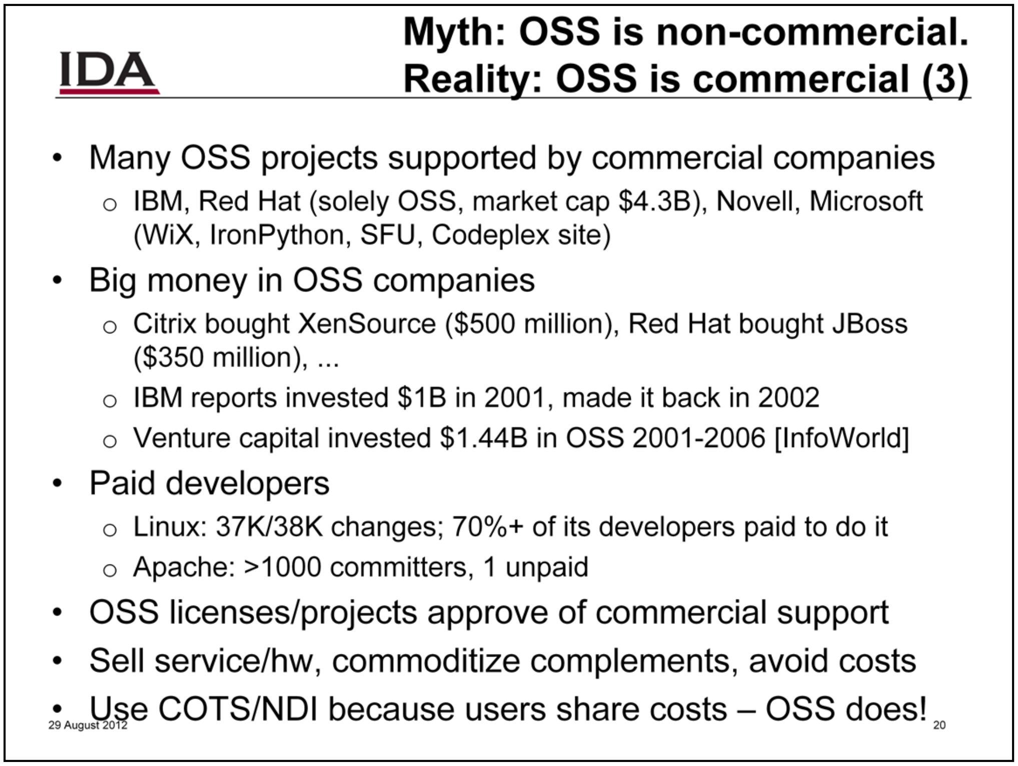
- Nearly all OSS are commercial items, & if extant, COTS
- U.S. Law (41 USC 403), FAR, & DFARS
  - Commercial item is “(1) Any item, other than real property, that is of a type customarily used by the general public or by non-governmental entities for purposes [not government-unique], and (i) Has been sold, leased, or licensed to the general public; or (ii) Has been offered for sale, lease, or license to the general public... (3) [Above with] (i) Modifications of a type customarily available in the commercial marketplace; or (ii) Minor modifications…”
  - Intentionally broad; "enables the Government to take greater advantage of the commercial marketplace” [DoD AT&L]
- Confirmed by DoD “Clarifying Guidance Regarding OSS” (Oct 16, 2009) & Navy “OSS Guidance” (June 5, 2007)



IDA

Myth: OSS is non-commercial. Reality: OSS is commercial (2)

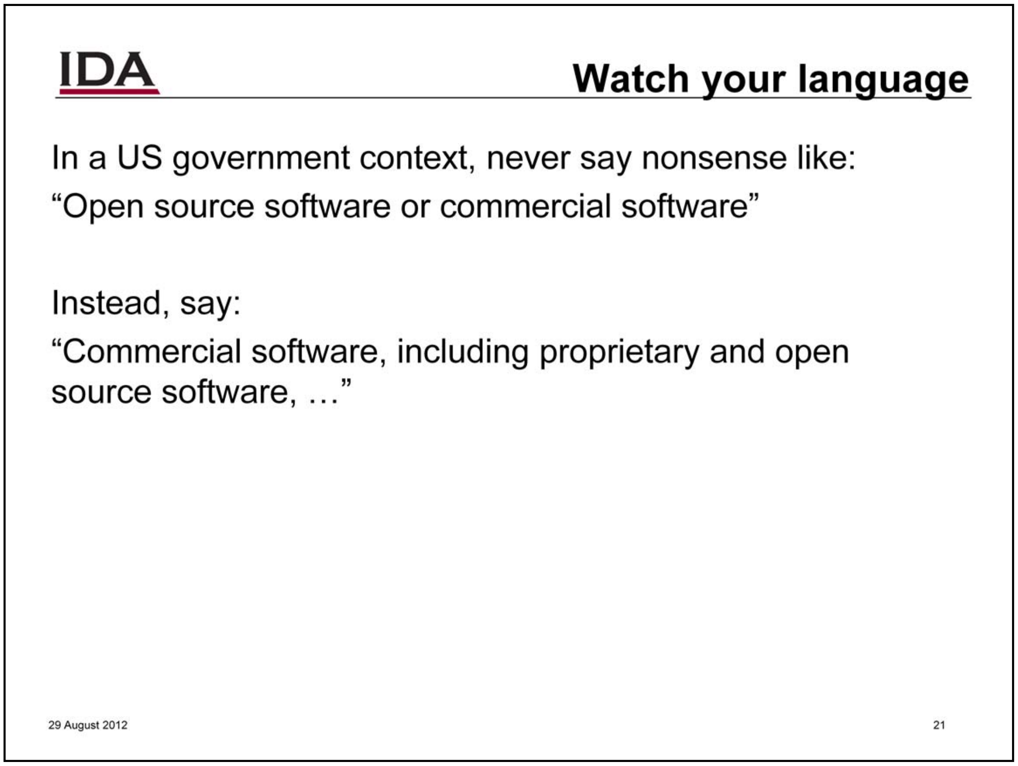
- OSS projects seek improvements = financial gain per
  - 17 USC 101: “financial gain” inc. “receipt, or expectation of receipt, of anything of value, including the receipt of other copyrighted works.”
- OMB Memo M-03-14: Commercial software, OSS support
- Important because U.S. Law (41 USC 403), FAR, DFARS require preference of commercial items (inc. COTS) & NDI:
  - Agencies must “(a) Conduct market research to determine [if] commercial items or nondevelopmental items are available … (b) Acquire [them when available] (c) Require prime contractors and subcontractors at all tiers to incorporate, to the maximum extent practicable, [them] as components...”



IDA

Myth: OSS is non-commercial.
Reality: OSS is commercial (3)

- Many OSS projects supported by commercial companies
  - IBM, Red Hat (solely OSS, market cap $4.3B), Novell, Microsoft (WiX, IronPython, SFU, Codeplex site)
- Big money in OSS companies
  - Citrix bought XenSource ($500 million), Red Hat bought JBoss ($350 million), ...
  - IBM reports invested $1B in 2001, made it back in 2002
  - Venture capital invested $1.44B in OSS 2001-2006 [InfoWorld]
- Paid developers
  - Linux: 37K/38K changes; 70%+ of its developers paid to do it
  - Apache: >1000 committers, 1 unpaid
- OSS licenses/projects approve of commercial support
- Sell service/hw, commoditize complements, avoid costs
- Use COTS/NDI because users share costs – OSS does!



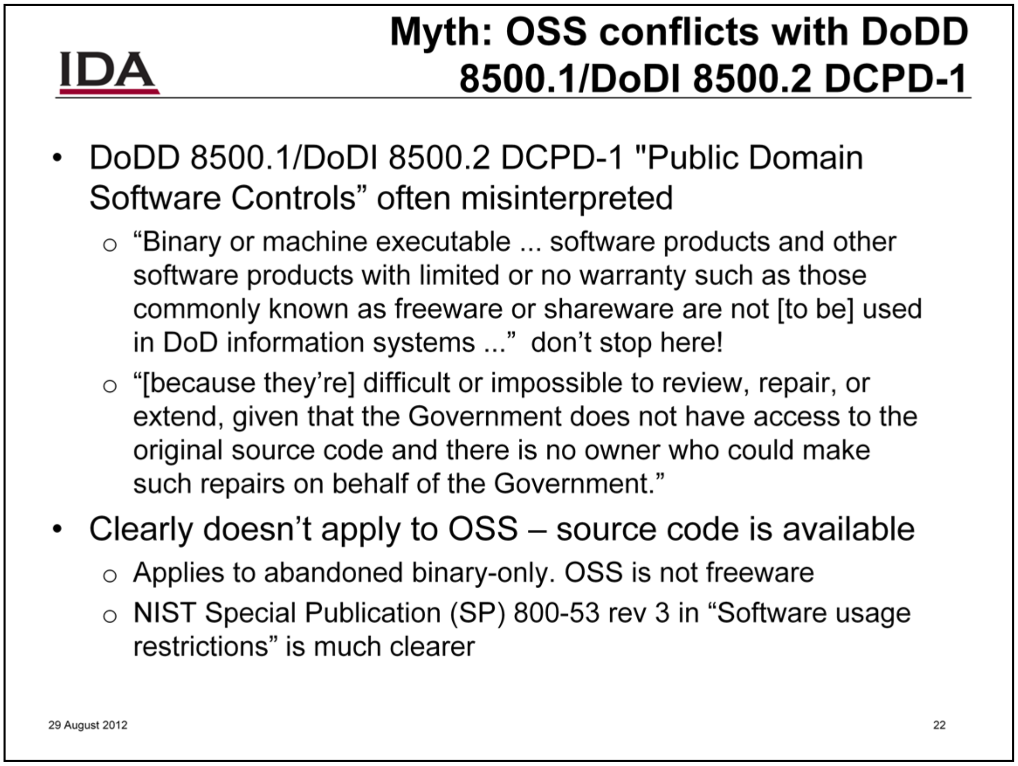
Watch your language

In a US government context, never say nonsense like:

“Open source software or commercial software”

Instead, say:

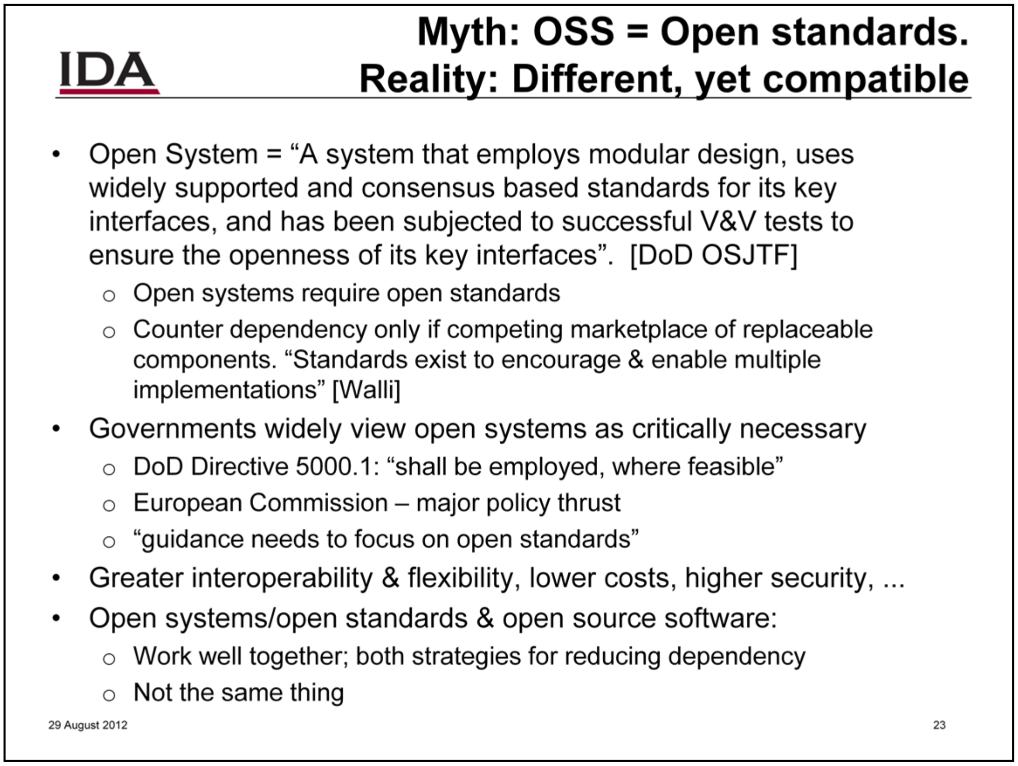
“Commercial software, including proprietary and open source software, …”



Myth: OSS conflicts with DoDD 8500.1/DoDI 8500.2 DCPD-1

- DoDD 8500.1/DoDI 8500.2 DCPD-1 "Public Domain Software Controls" often misinterpreted
  - "Binary or machine executable ... software products and other software products with limited or no warranty such as those commonly known as freeware or shareware are not [to be] used in DoD information systems ..." don't stop here!
  - "[because they're] difficult or impossible to review, repair, or extend, given that the Government does not have access to the original source code and there is no owner who could make such repairs on behalf of the Government."
- Clearly doesn't apply to OSS – source code is available
  - Applies to abandoned binary-only. OSS is not freeware
  - NIST Special Publication (SP) 800-53 rev 3 in "Software usage restrictions" is much clearer

29 August 2012 22

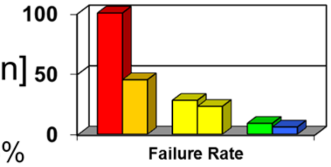


IDA

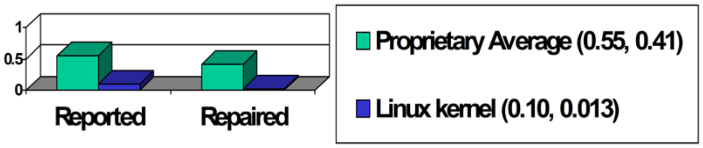
Myth: OSS = Open standards. Reality: Different, yet compatible

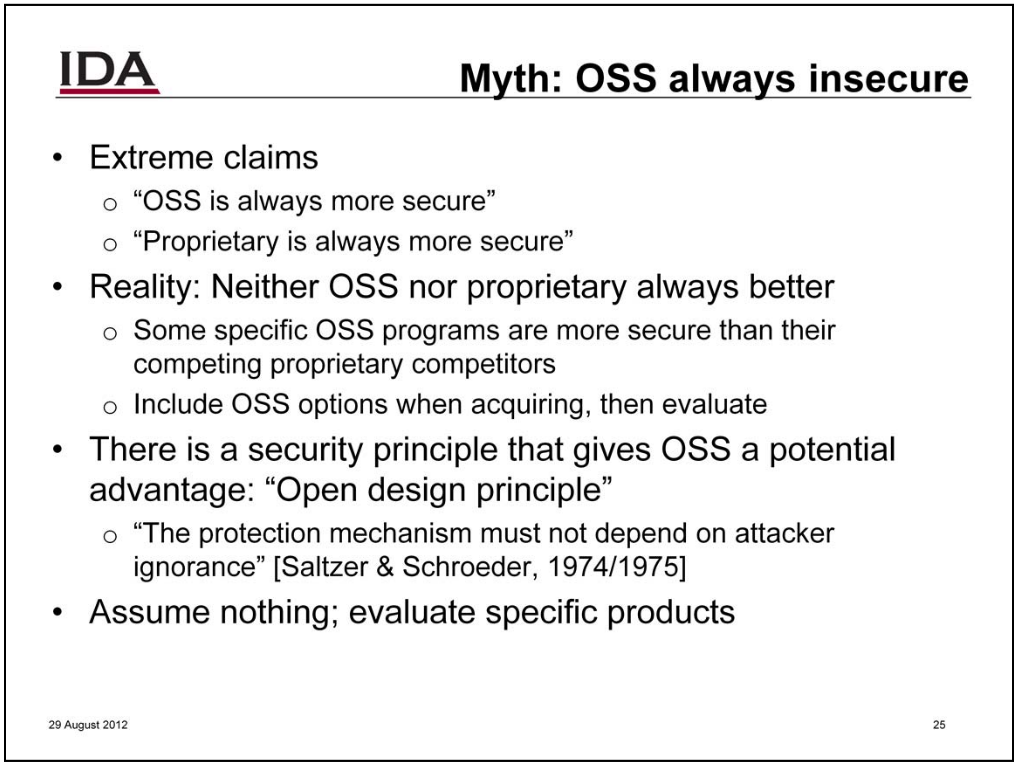
- Open System = "A system that employs modular design, uses widely supported and consensus based standards for its key interfaces, and has been subjected to successful V&V tests to ensure the openness of its key interfaces". [DoD OSJTF]
  - Open systems require open standards
  - Counter dependency only if competing marketplace of replaceable components. "Standards exist to encourage & enable multiple implementations" [Walli]
- Governments widely view open systems as critically necessary
  - DoD Directive 5000.1: "shall be employed, where feasible"
  - European Commission – major policy thrust
  - "guidance needs to focus on open standards"
- Greater interoperability & flexibility, lower costs, higher security, ...
- Open systems/open standards & open source software:
  - Work well together; both strategies for reducing dependency
  - Not the same thing

Myth: OSS always unreliable
Reality: OSS often very reliable



- Fuzz studies found OSS apps significantly more reliable [U Wisconsin]
  - Proprietary Unix failure rate: 28%,23%
  - OSS: Slackware Linux 9%, GNU utilities 6%
  - Windows: 100%; 45% if forbid certain Win32 message formats
- IIS web servers >2x downtime of Apache [Syscontrol AG]
- Linux kernel TCP/IP had smaller defect density [Reasoning]

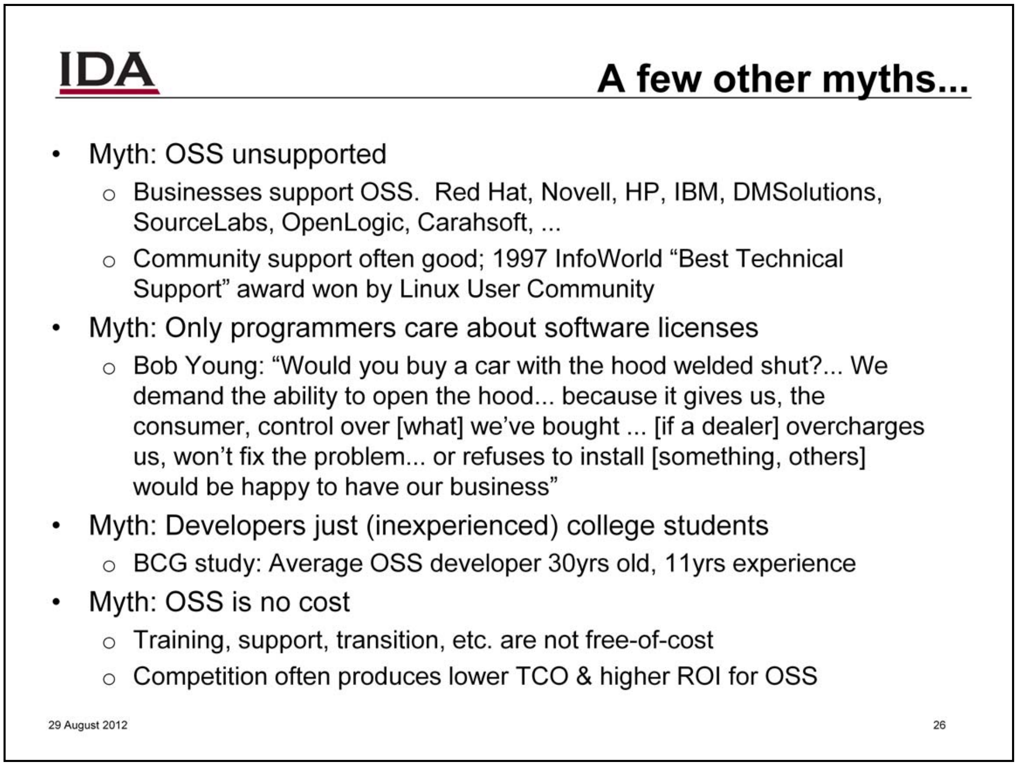




IDA

Myth: OSS always insecure

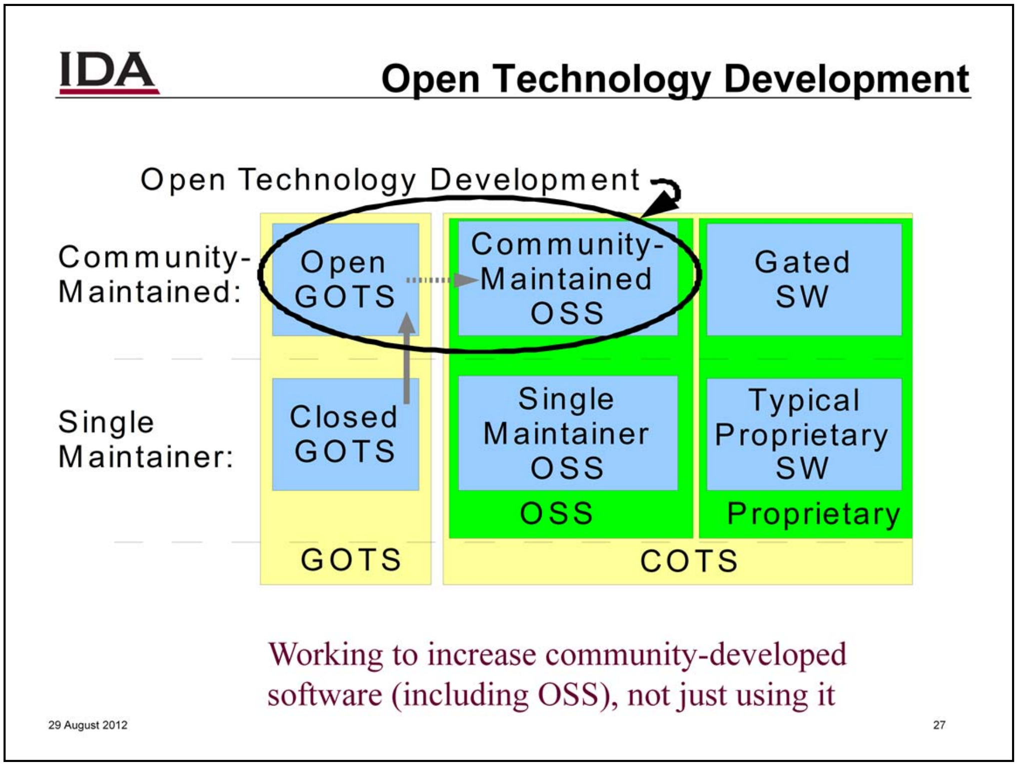
- Extreme claims
  - “OSS is always more secure”
  - “Proprietary is always more secure”
- Reality: Neither OSS nor proprietary always better
  - Some specific OSS programs are more secure than their competing proprietary competitors
  - Include OSS options when acquiring, then evaluate
- There is a security principle that gives OSS a potential advantage: “Open design principle”
  - “The protection mechanism must not depend on attacker ignorance” [Saltzer & Schroeder, 1974/1975]
- Assume nothing; evaluate specific products



IDA

A few other myths...

- Myth: OSS unsupported
  - Businesses support OSS. Red Hat, Novell, HP, IBM, DMSolutions, SourceLabs, OpenLogic, Carahsoft, ...
  - Community support often good; 1997 InfoWorld “Best Technical Support” award won by Linux User Community
- Myth: Only programmers care about software licenses
  - Bob Young: “Would you buy a car with the hood welded shut?... We demand the ability to open the hood... because it gives us, the consumer, control over [what] we’ve bought ... [if a dealer] overcharges us, won’t fix the problem... or refuses to install [something, others] would be happy to have our business”
- Myth: Developers just (inexperienced) college students
  - BCG study: Average OSS developer 30yrs old, 11yrs experience
- Myth: OSS is no cost
  - Training, support, transition, etc. are not free-of-cost
  - Competition often produces lower TCO & higher ROI for OSS



IDA

Open Technology Development

Open Technology Development

	GOTS	COTS	
		OSS	Proprietary
Community-Maintained:	Open GOTS	Community-Maintained OSS	Gated SW
Single Maintainer:	Closed GOTS	Single Maintainer OSS	Typical Proprietary SW

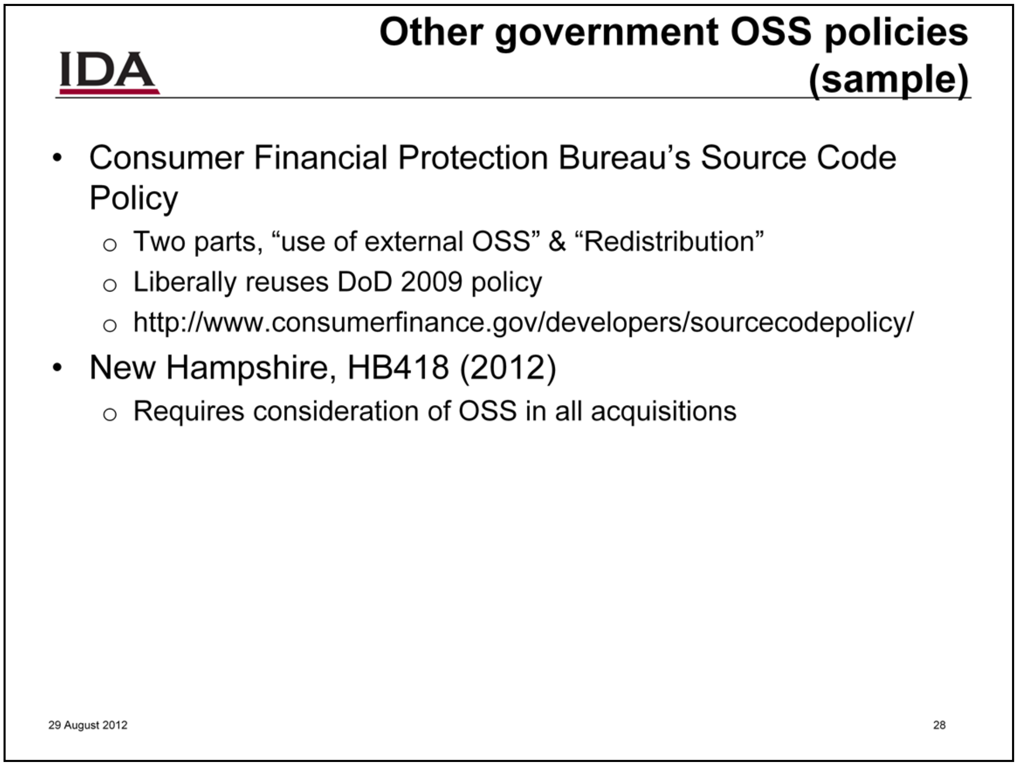
Working to increase community-developed software (including OSS), not just using it

29 August 2012

27

This figure is from “Open Technology Development (OTD): Lessons Learned & Best Practices for Military Software”, OSD Report, May 2011, http://dodcio.defense.gov/Portals/0/Documents/FOSS/OTD-lessons-learned-military-signed.pdf

No hypertext linking policy found.



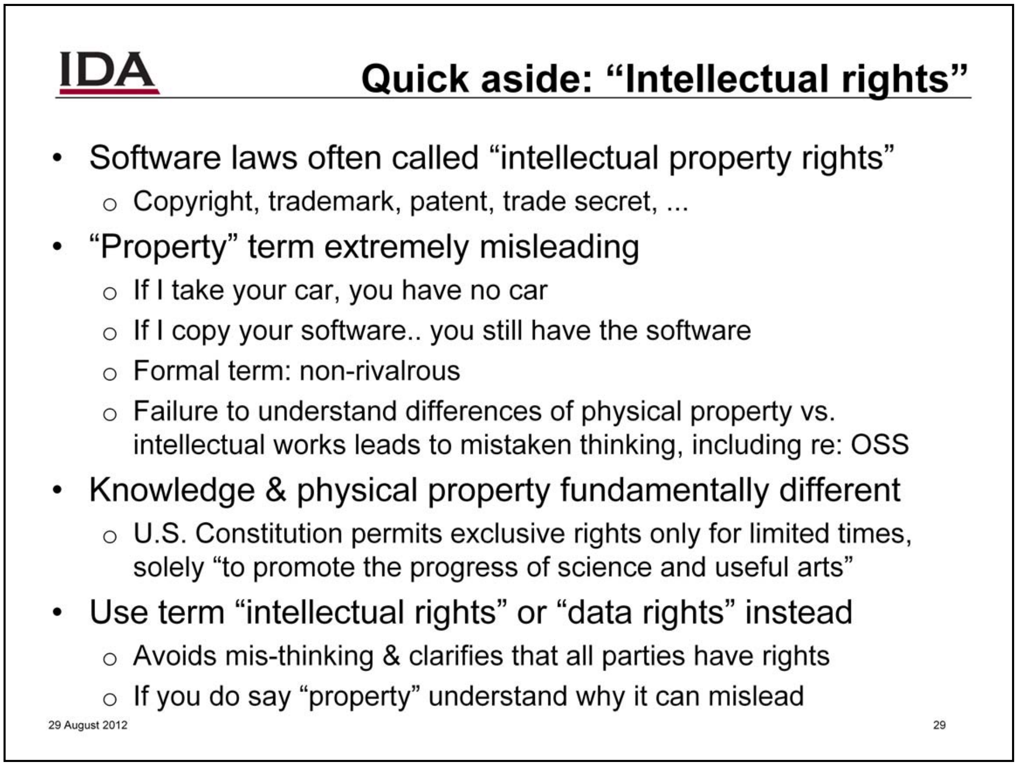
IDA

Other government OSS policies (sample)

- Consumer Financial Protection Bureau’s Source Code Policy
  - Two parts, “use of external OSS” & “Redistribution”
  - Liberally reuses DoD 2009 policy
  - http://www.consumerfinance.gov/developers/sourcecodepolicy/
- New Hampshire, HB418 (2012)
  - Requires consideration of OSS in all acquisitions

29 August 2012 28

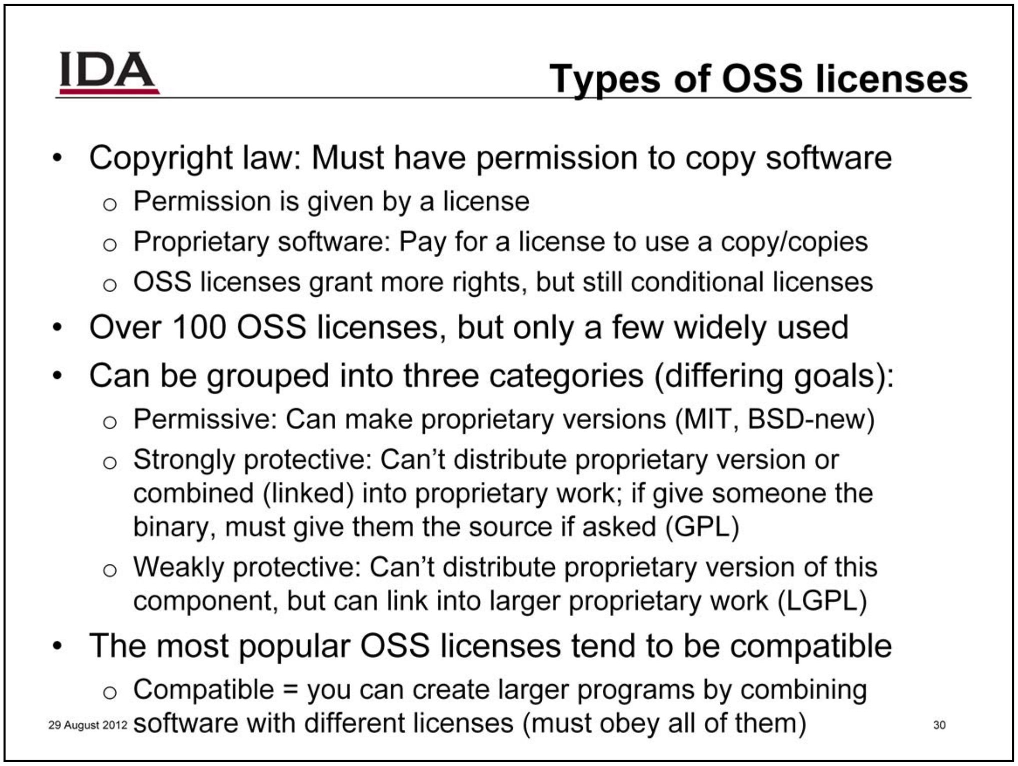
No hypertext linking policy was found starting from:
http://www.consumerfinance.gov/developers/sourcecodepolicy



IDA

Quick aside: “Intellectual rights”

- Software laws often called “intellectual property rights”
  - Copyright, trademark, patent, trade secret, ...
- “Property” term extremely misleading
  - If I take your car, you have no car
  - If I copy your software.. you still have the software
  - Formal term: non-rivalrous
  - Failure to understand differences of physical property vs. intellectual works leads to mistaken thinking, including re: OSS
- Knowledge & physical property fundamentally different
  - U.S. Constitution permits exclusive rights only for limited times, solely “to promote the progress of science and useful arts”
- Use term “intellectual rights” or “data rights” instead
  - Avoids mis-thinking & clarifies that all parties have rights
  - If you do say “property” understand why it can mislead

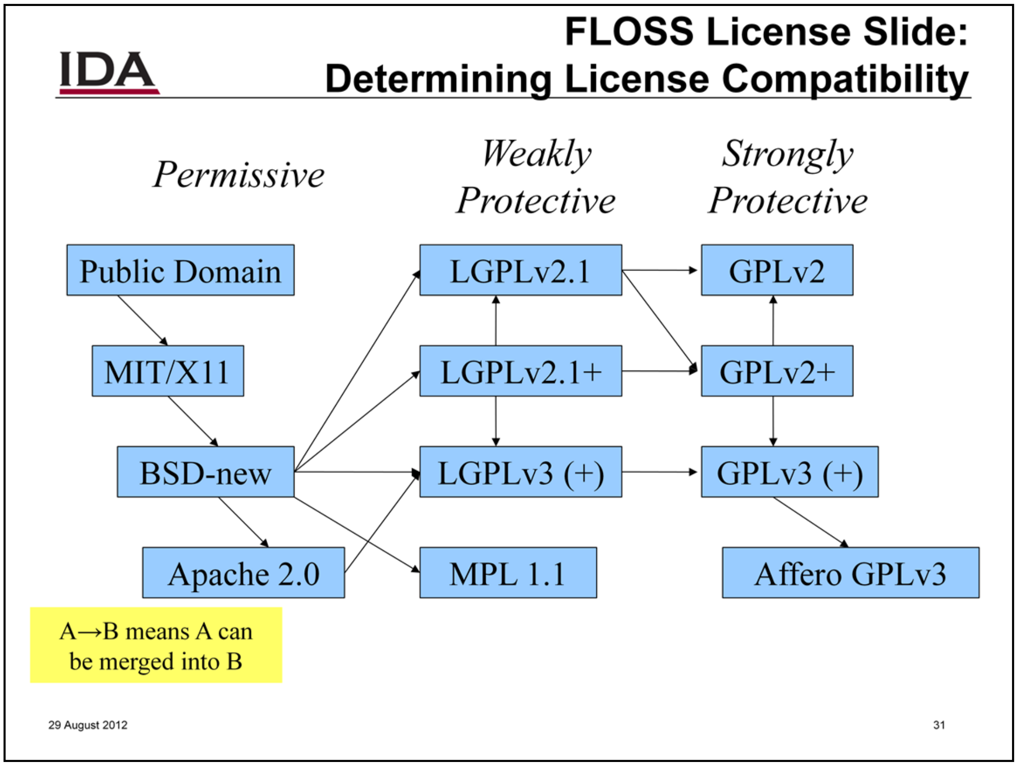


IDA

Types of OSS licenses

- Copyright law: Must have permission to copy software
  - Permission is given by a license
  - Proprietary software: Pay for a license to use a copy/copies
  - OSS licenses grant more rights, but still conditional licenses
- Over 100 OSS licenses, but only a few widely used
- Can be grouped into three categories (differing goals):
  - Permissive: Can make proprietary versions (MIT, BSD-new)
  - Strongly protective: Can't distribute proprietary version or combined (linked) into proprietary work; if give someone the binary, must give them the source if asked (GPL)
  - Weakly protective: Can't distribute proprietary version of this component, but can link into larger proprietary work (LGPL)
- The most popular OSS licenses tend to be compatible
  - Compatible = you can create larger programs by combining software with different licenses (must obey all of them)

29 August 2012 30



IDA

FLOSS License Slide: Determining License Compatibility

Permissive

Weakly Protective

Strongly Protective

Public Domain

MIT/X11

BSD-new

Apache 2.0

LGPLv2.1

LGPLv2.1+

LGPLv3 (+)

MPL 1.1

GPLv2

GPLv2+

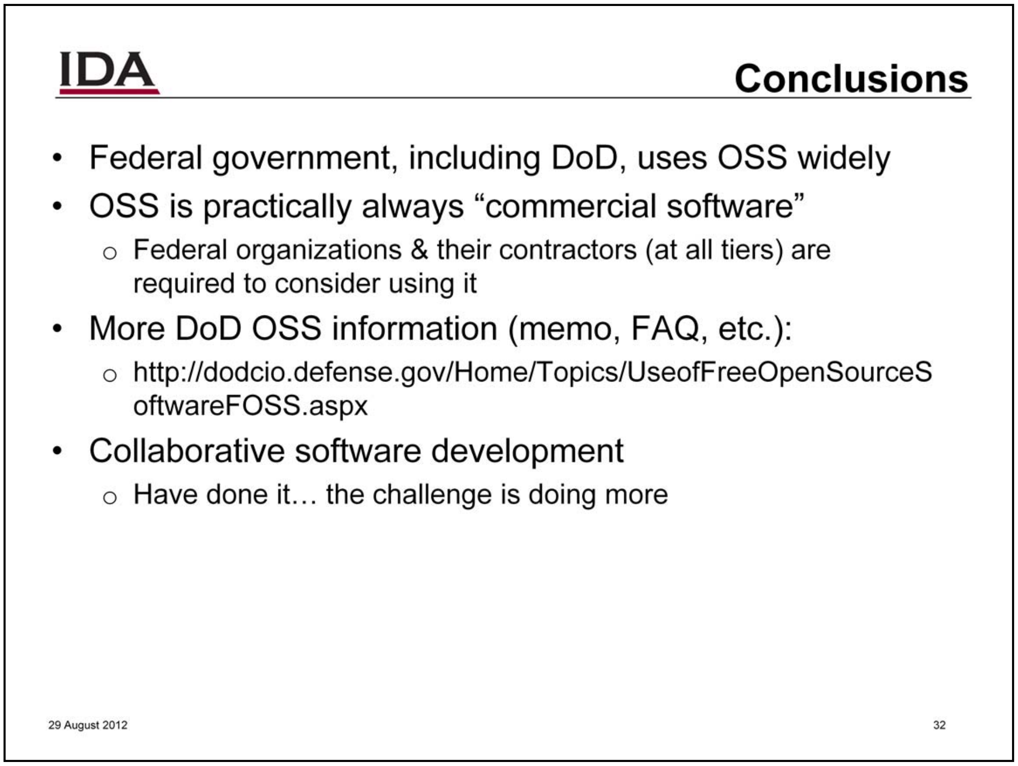
GPLv3 (+)

Affero GPLv3

A→B means A can be merged into B

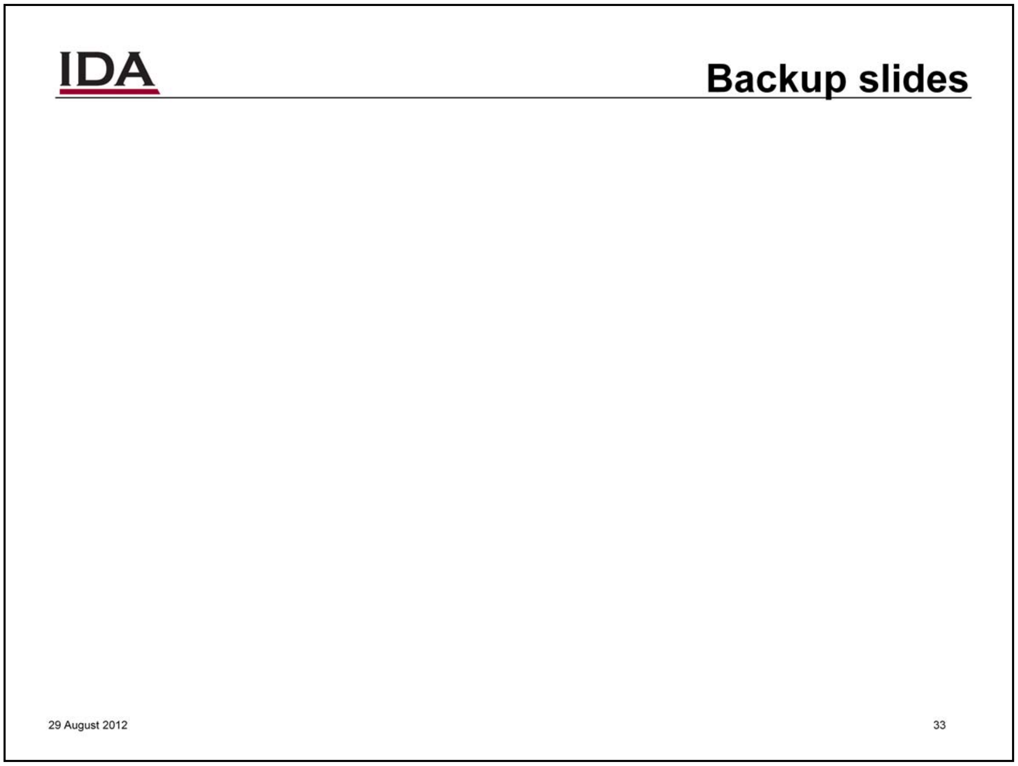
29 August 2012

31



Conclusions

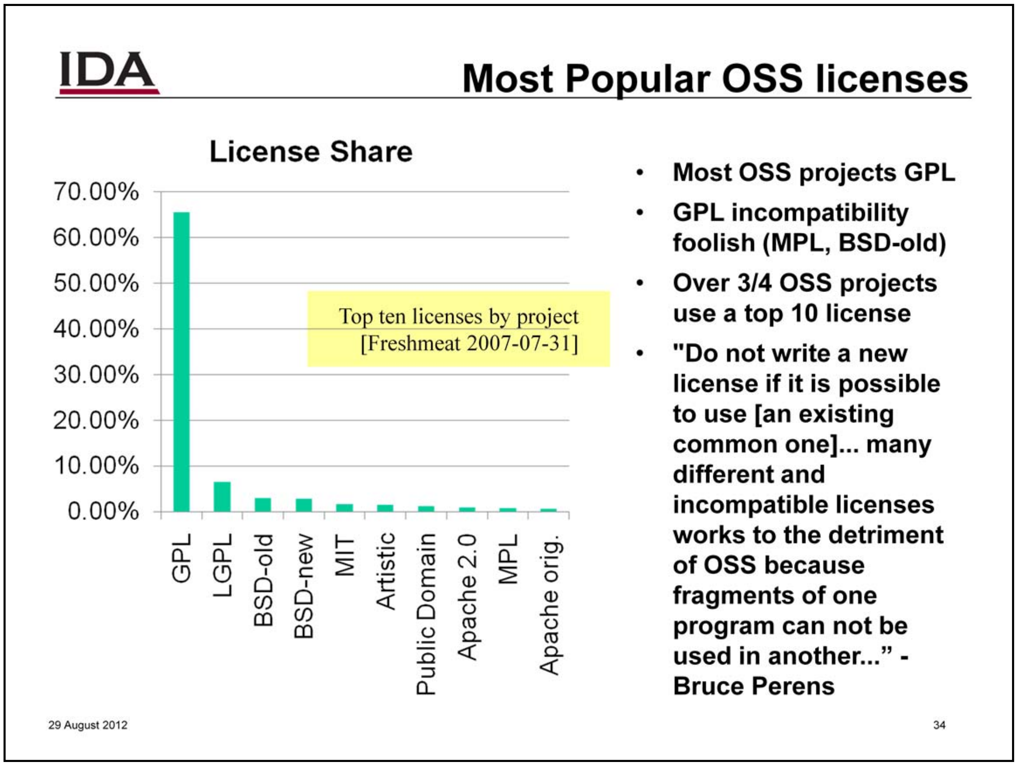
- Federal government, including DoD, uses OSS widely
- OSS is practically always “commercial software”
  - Federal organizations & their contractors (at all tiers) are required to consider using it
- More DoD OSS information (memo, FAQ, etc.):
  - http://dodcio.defense.gov/Home/Topics/UseofFreeOpenSourceSoftwareFOSS.aspx
- Collaborative software development
  - Have done it… the challenge is doing more



Backup slides

29 August 2012

33

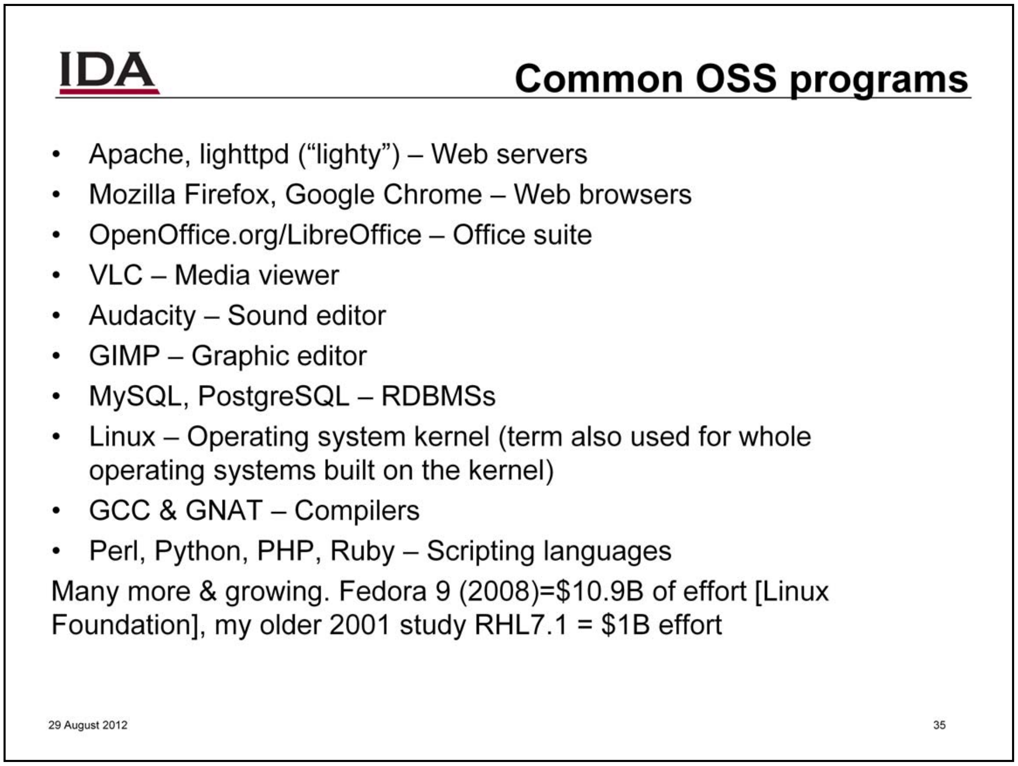


IDA

Most Popular OSS licenses

Top ten licenses by project [Freshmeat 2007-07-31]

- Most OSS projects GPL
- GPL incompatibility foolish (MPL, BSD-old)
- Over 3/4 OSS projects use a top 10 license
- "Do not write a new license if it is possible to use [an existing common one]... many different and incompatible licenses works to the detriment of OSS because fragments of one program can not be used in another...” - Bruce Perens

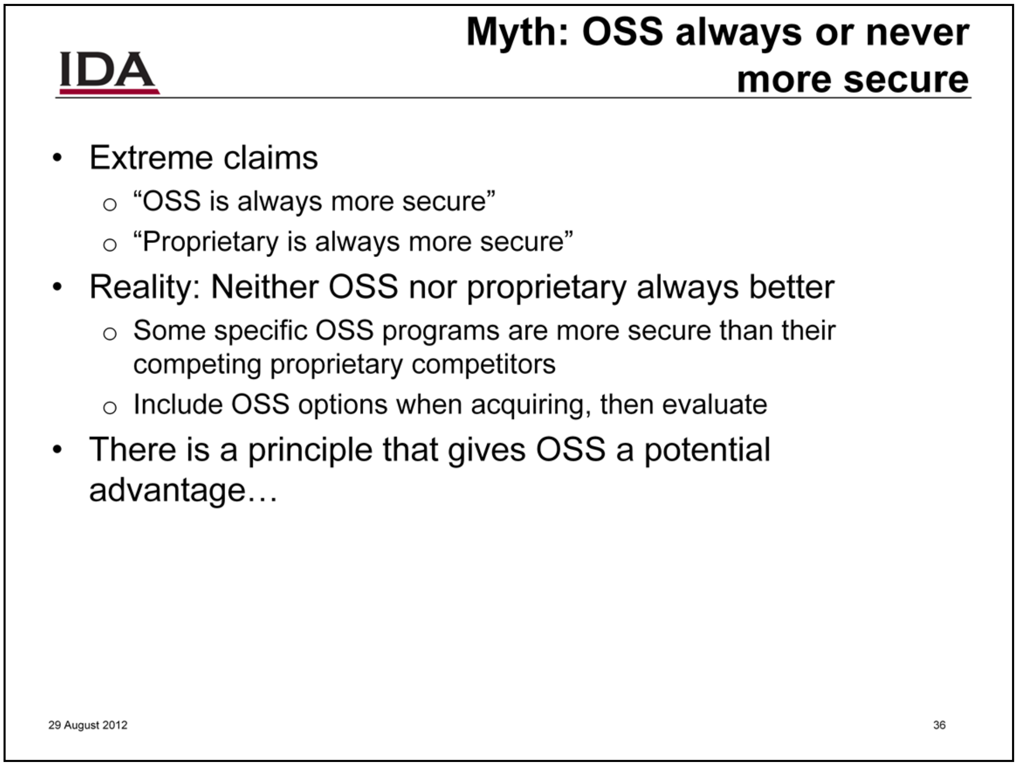


IDA

Common OSS programs

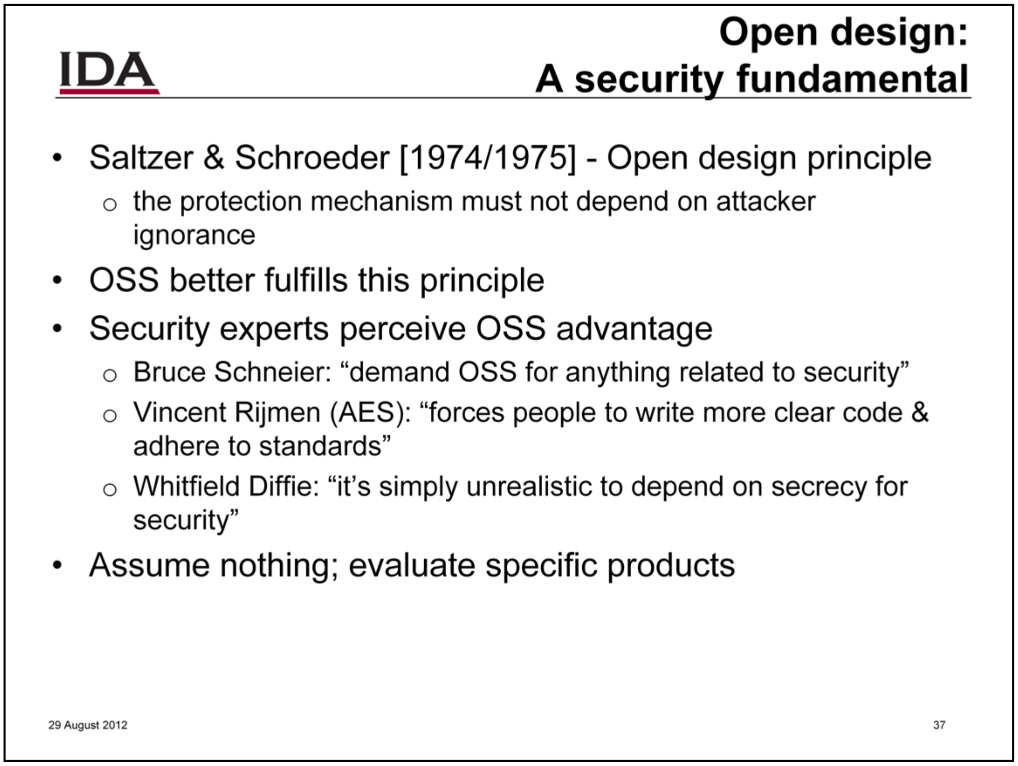
- Apache, lighttpd (“lighty”) – Web servers
- Mozilla Firefox, Google Chrome – Web browsers
- OpenOffice.org/LibreOffice – Office suite
- VLC – Media viewer
- Audacity – Sound editor
- GIMP – Graphic editor
- MySQL, PostgreSQL – RDBMSs
- Linux – Operating system kernel (term also used for whole operating systems built on the kernel)
- GCC & GNAT – Compilers
- Perl, Python, PHP, Ruby – Scripting languages

Many more & growing. Fedora 9 (2008)=$10.9B of effort [Linux Foundation], my older 2001 study RHL7.1 = $1B effort



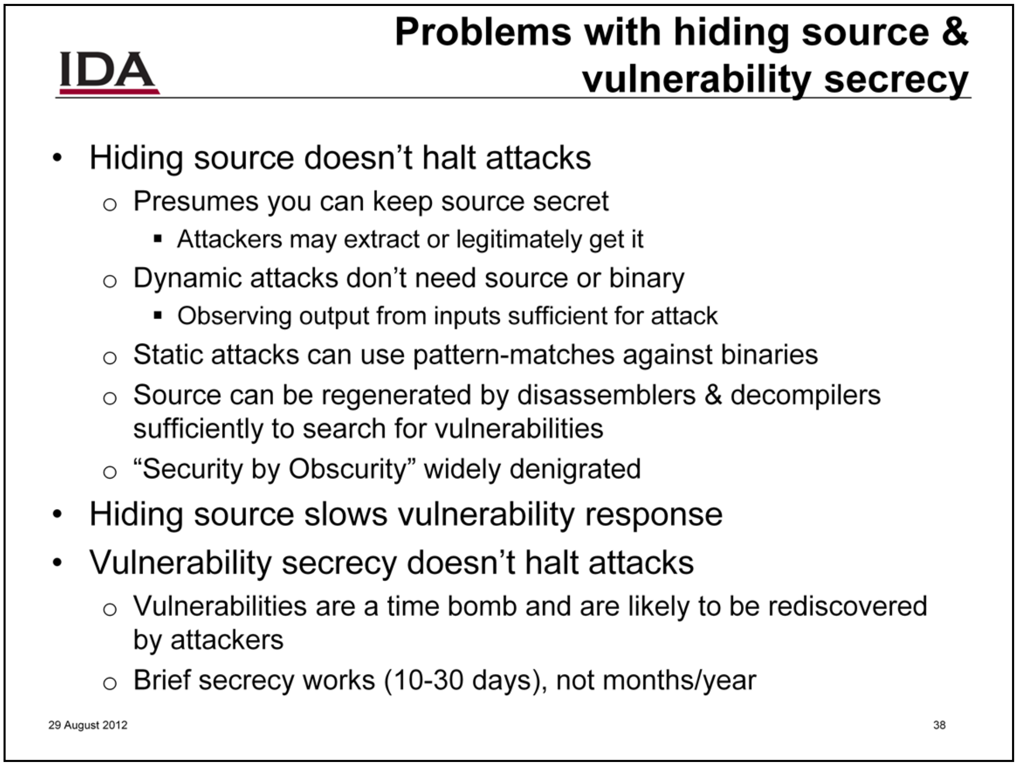
Myth: OSS always or never more secure

- Extreme claims
  - “OSS is always more secure”
  - “Proprietary is always more secure”
- Reality: Neither OSS nor proprietary always better
  - Some specific OSS programs are more secure than their competing proprietary competitors
  - Include OSS options when acquiring, then evaluate
- There is a principle that gives OSS a potential advantage…



Open design: A security fundamental

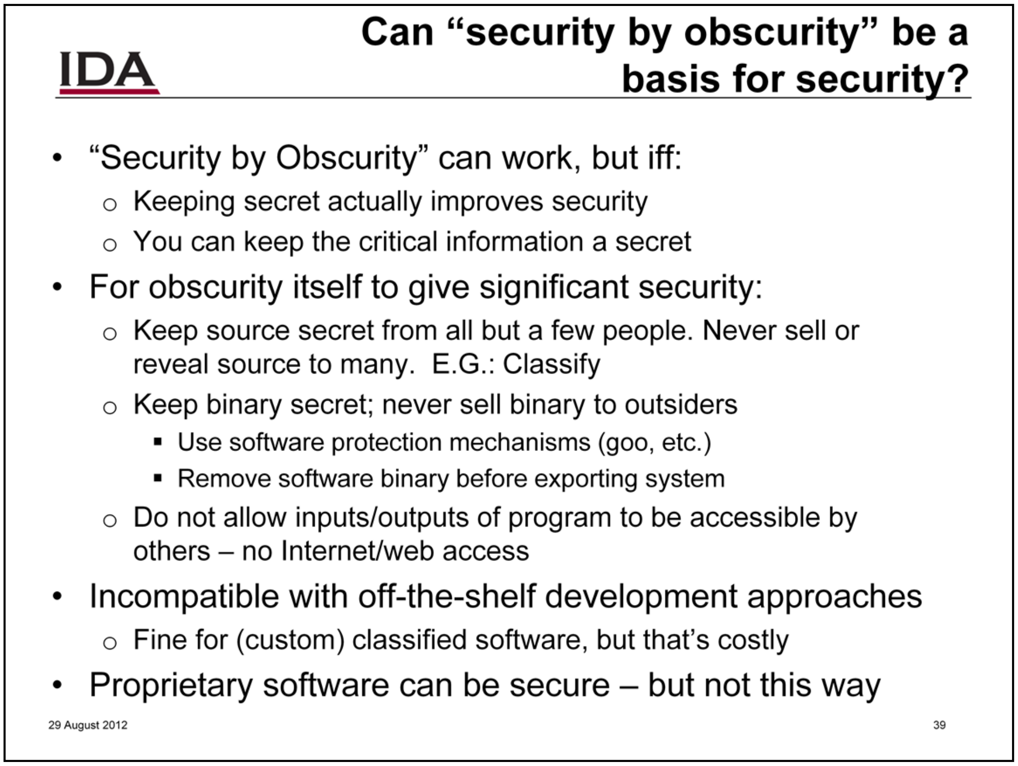
- Saltzer & Schroeder [1974/1975] - Open design principle
  - the protection mechanism must not depend on attacker ignorance
- OSS better fulfills this principle
- Security experts perceive OSS advantage
  - Bruce Schneier: “demand OSS for anything related to security”
  - Vincent Rijmen (AES): “forces people to write more clear code & adhere to standards”
  - Whitfield Diffie: “it’s simply unrealistic to depend on secrecy for security”
- Assume nothing; evaluate specific products



IDA

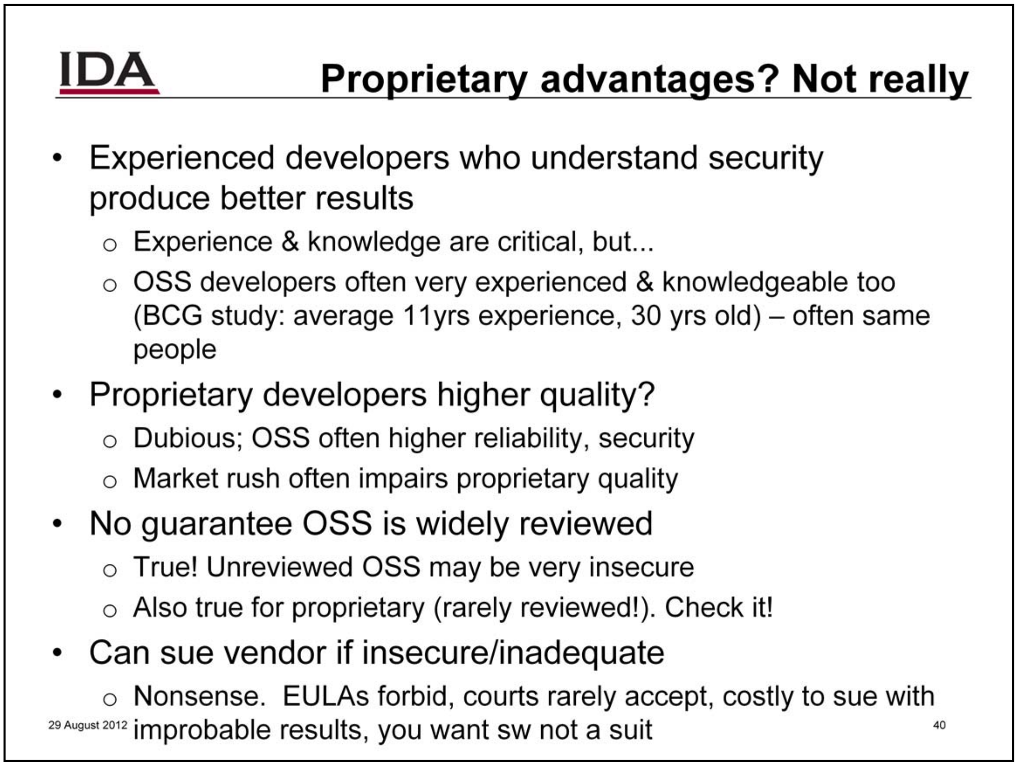
Problems with hiding source & vulnerability secrecy

- Hiding source doesn't halt attacks
  - Presumes you can keep source secret
    - Attackers may extract or legitimately get it
  - Dynamic attacks don't need source or binary
    - Observing output from inputs sufficient for attack
  - Static attacks can use pattern-matches against binaries
  - Source can be regenerated by disassemblers & decompilers sufficiently to search for vulnerabilities
  - "Security by Obscurity" widely denigrated
- Hiding source slows vulnerability response
- Vulnerability secrecy doesn't halt attacks
  - Vulnerabilities are a time bomb and are likely to be rediscovered by attackers
  - Brief secrecy works (10-30 days), not months/year



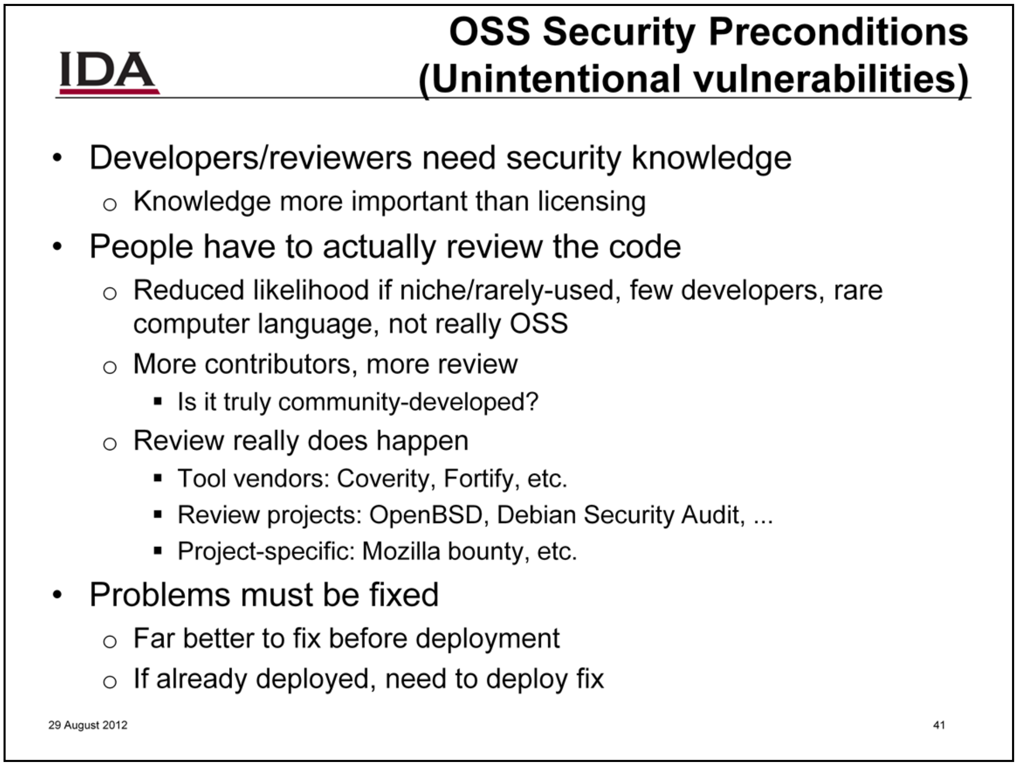
Can "security by obscurity" be a basis for security?

- "Security by Obscurity" can work, but iff:
  - Keeping secret actually improves security
  - You can keep the critical information a secret
- For obscurity itself to give significant security:
  - Keep source secret from all but a few people. Never sell or reveal source to many. E.G.: Classify
  - Keep binary secret; never sell binary to outsiders
    - Use software protection mechanisms (goo, etc.)
    - Remove software binary before exporting system
  - Do not allow inputs/outputs of program to be accessible by others – no Internet/web access
- Incompatible with off-the-shelf development approaches
  - Fine for (custom) classified software, but that's costly
- Proprietary software can be secure – but not this way



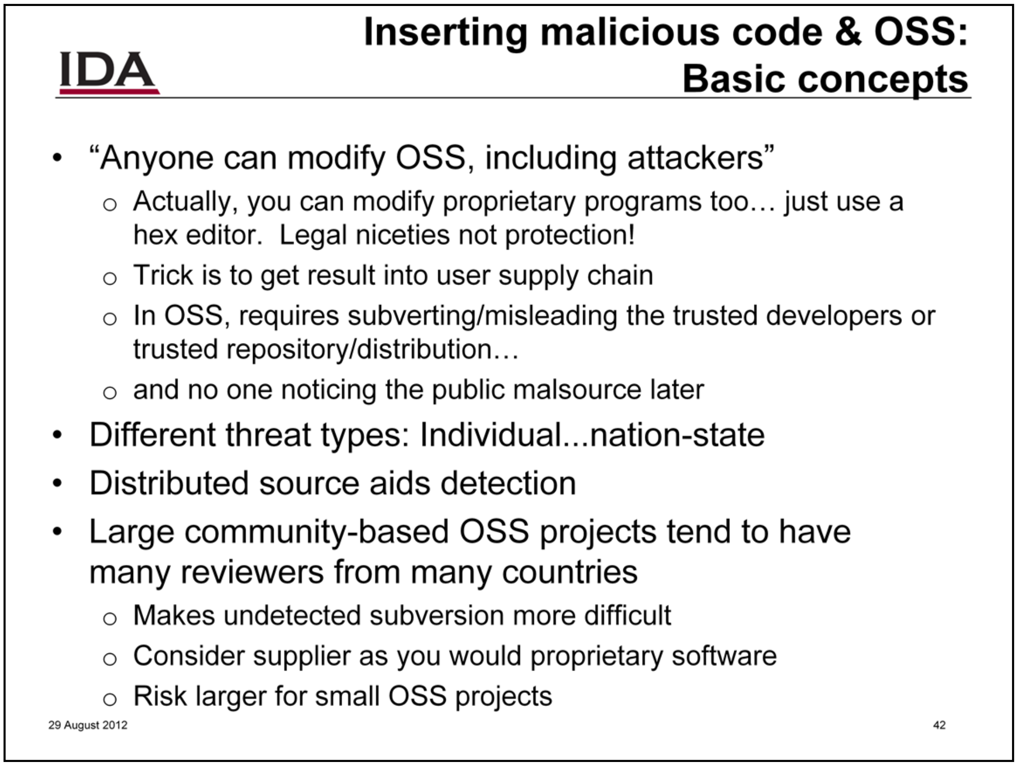
Proprietary advantages? Not really

- Experienced developers who understand security produce better results
  - Experience & knowledge are critical, but...
  - OSS developers often very experienced & knowledgeable too (BCG study: average 11yrs experience, 30 yrs old) – often same people
- Proprietary developers higher quality?
  - Dubious; OSS often higher reliability, security
  - Market rush often impairs proprietary quality
- No guarantee OSS is widely reviewed
  - True! Unreviewed OSS may be very insecure
  - Also true for proprietary (rarely reviewed!). Check it!
- Can sue vendor if insecure/inadequate
  - Nonsense. EULAs forbid, courts rarely accept, costly to sue with improbable results, you want sw not a suit



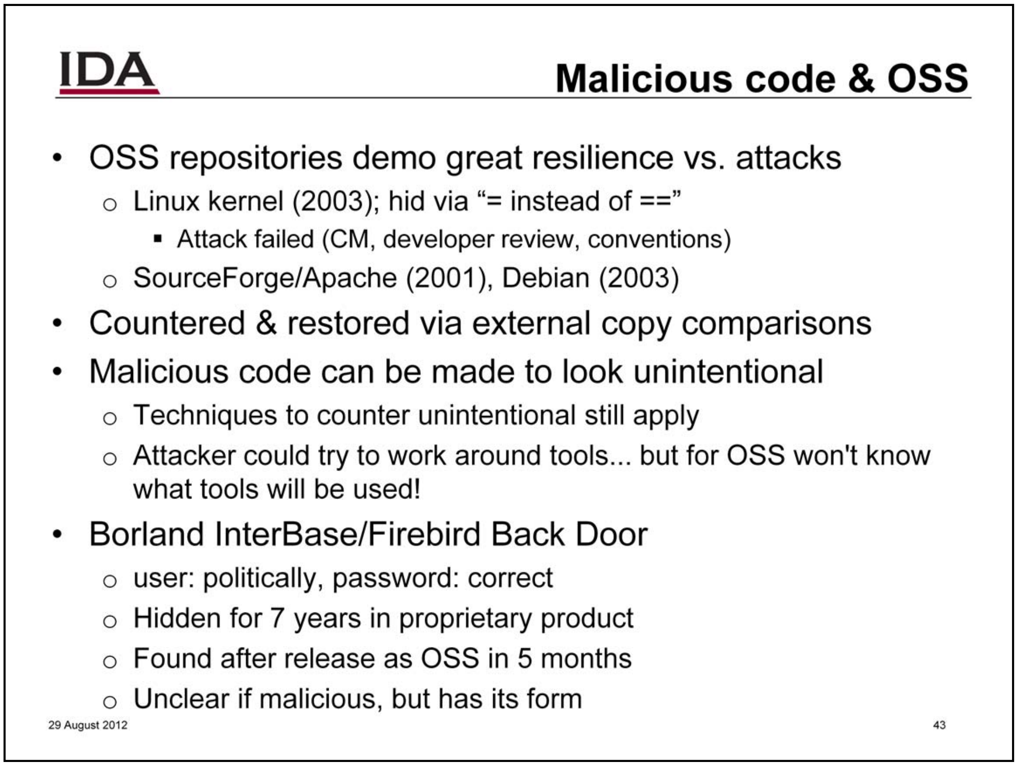
OSS Security Preconditions (Unintentional vulnerabilities)

- Developers/reviewers need security knowledge
  - Knowledge more important than licensing
- People have to actually review the code
  - Reduced likelihood if niche/rarely-used, few developers, rare computer language, not really OSS
  - More contributors, more review
    - Is it truly community-developed?
  - Review really does happen
    - Tool vendors: Coverity, Fortify, etc.
    - Review projects: OpenBSD, Debian Security Audit, ...
    - Project-specific: Mozilla bounty, etc.
- Problems must be fixed
  - Far better to fix before deployment
  - If already deployed, need to deploy fix



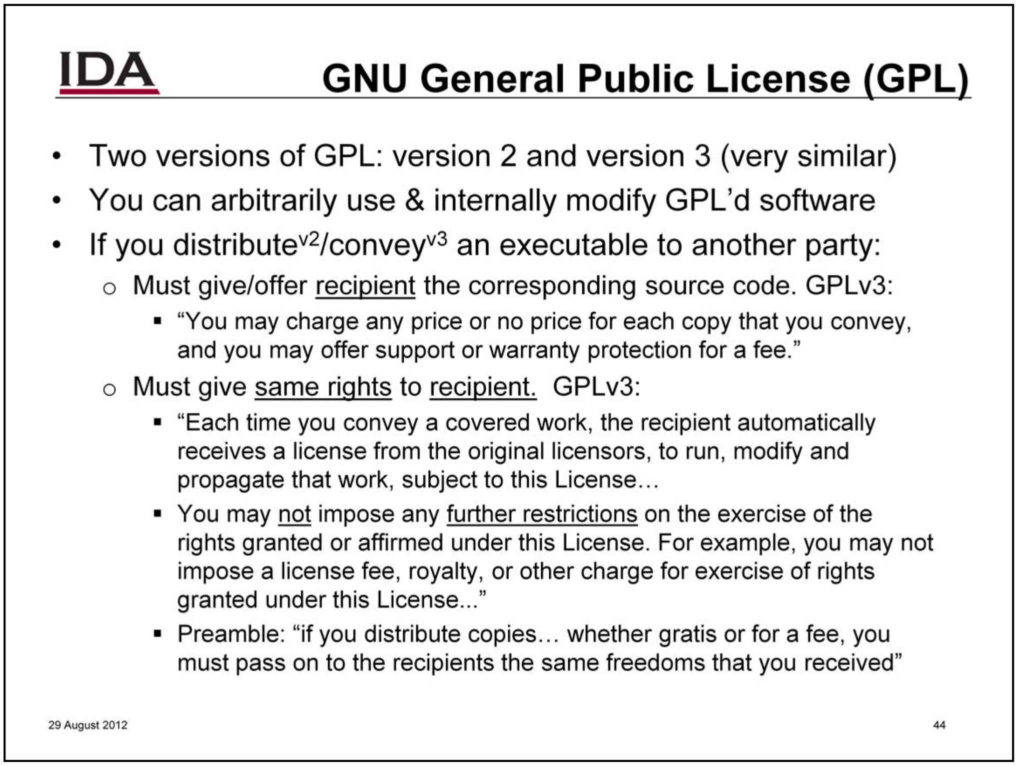
Inserting malicious code & OSS: Basic concepts

- “Anyone can modify OSS, including attackers”
  - Actually, you can modify proprietary programs too… just use a hex editor. Legal niceties not protection!
  - Trick is to get result into user supply chain
  - In OSS, requires subverting/misleading the trusted developers or trusted repository/distribution…
  - and no one noticing the public malsource later
- Different threat types: Individual...nation-state
- Distributed source aids detection
- Large community-based OSS projects tend to have many reviewers from many countries
  - Makes undetected subversion more difficult
  - Consider supplier as you would proprietary software
  - Risk larger for small OSS projects



Malicious code & OSS

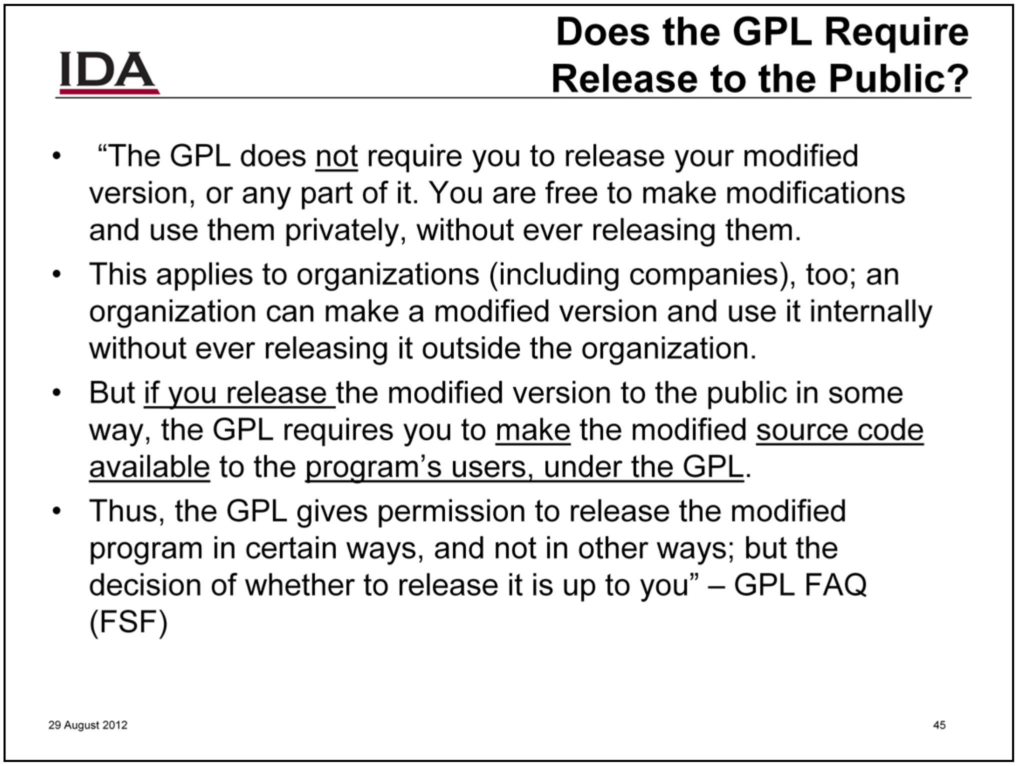
- OSS repositories demo great resilience vs. attacks
  - Linux kernel (2003); hid via “= instead of ==”
    - Attack failed (CM, developer review, conventions)
  - SourceForge/Apache (2001), Debian (2003)
- Countered & restored via external copy comparisons
- Malicious code can be made to look unintentional
  - Techniques to counter unintentional still apply
  - Attacker could try to work around tools... but for OSS won't know what tools will be used!
- Borland InterBase/Firebird Back Door
  - user: politically, password: correct
  - Hidden for 7 years in proprietary product
  - Found after release as OSS in 5 months
  - Unclear if malicious, but has its form



IDA

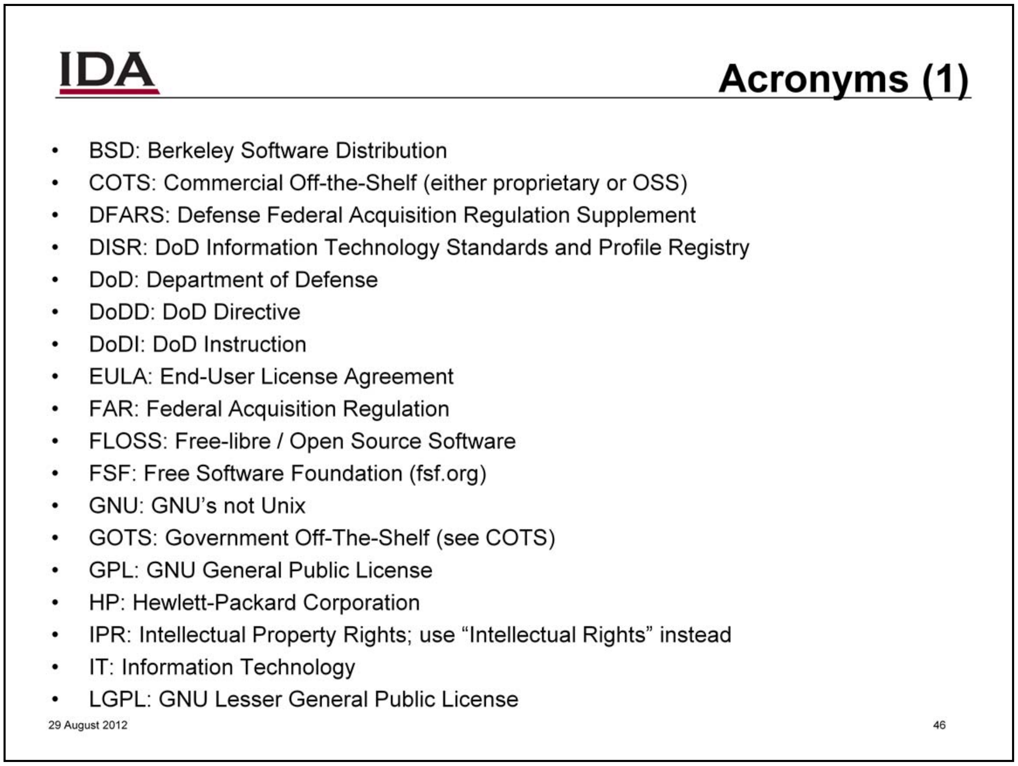
GNU General Public License (GPL)

- Two versions of GPL: version 2 and version 3 (very similar)
- You can arbitrarily use & internally modify GPL’d software
- If you distribute^{v2}/convey^{v3} an executable to another party:
  - Must give/offer recipient the corresponding source code. GPLv3:
    - “You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.”
  - Must give same rights to recipient. GPLv3:
    - “Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License…
    - You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License...”
    - Preamble: “if you distribute copies… whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received”



Does the GPL Require Release to the Public?

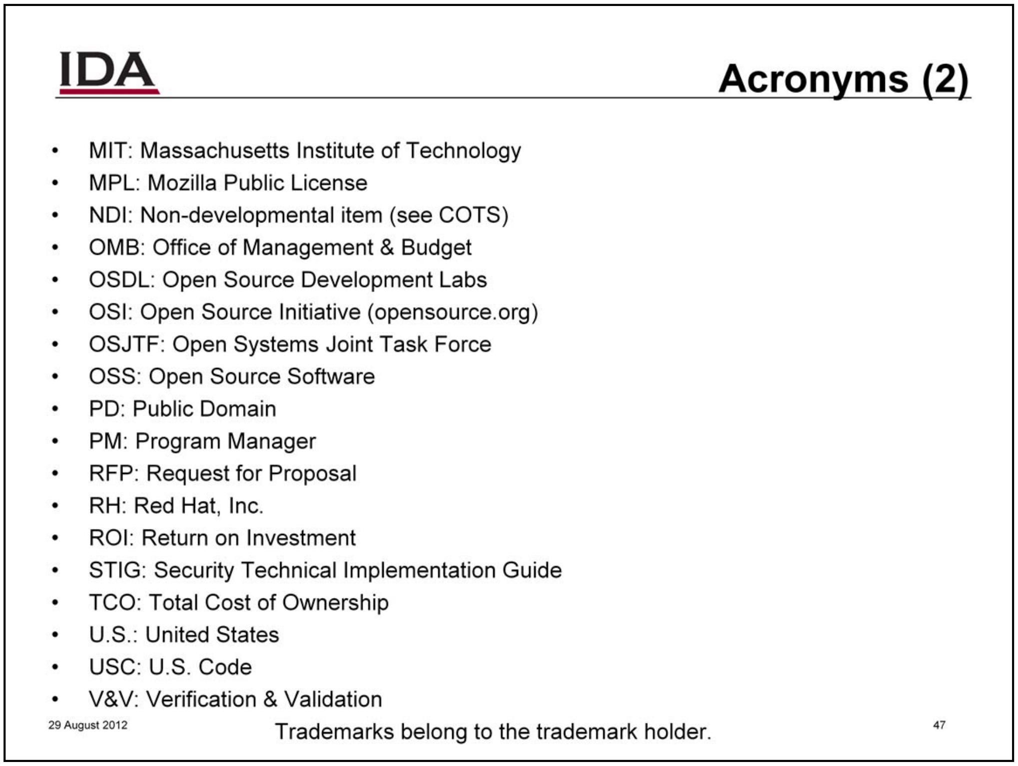
- “The GPL does not require you to release your modified version, or any part of it. You are free to make modifications and use them privately, without ever releasing them.
- This applies to organizations (including companies), too; an organization can make a modified version and use it internally without ever releasing it outside the organization.
- But if you release the modified version to the public in some way, the GPL requires you to make the modified source code available to the program’s users, under the GPL.
- Thus, the GPL gives permission to release the modified program in certain ways, and not in other ways; but the decision of whether to release it is up to you” – GPL FAQ (FSF)



IDA

Acronyms (1)

- BSD: Berkeley Software Distribution
- COTS: Commercial Off-the-Shelf (either proprietary or OSS)
- DFARS: Defense Federal Acquisition Regulation Supplement
- DISR: DoD Information Technology Standards and Profile Registry
- DoD: Department of Defense
- DoDD: DoD Directive
- DoDI: DoD Instruction
- EULA: End-User License Agreement
- FAR: Federal Acquisition Regulation
- FLOSS: Free-libre / Open Source Software
- FSF: Free Software Foundation (fsf.org)
- GNU: GNU’s not Unix
- GOTS: Government Off-The-Shelf (see COTS)
- GPL: GNU General Public License
- HP: Hewlett-Packard Corporation
- IPR: Intellectual Property Rights; use “Intellectual Rights” instead
- IT: Information Technology
- LGPL: GNU Lesser General Public License

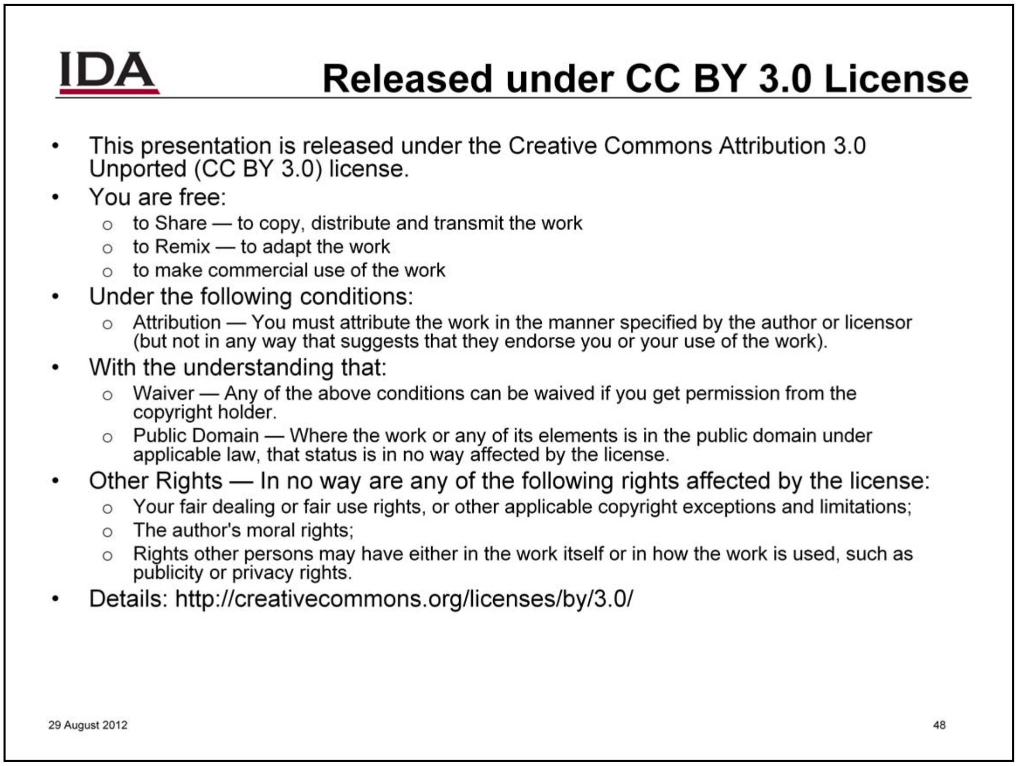


IDA

Acronyms (2)

- MIT: Massachusetts Institute of Technology
- MPL: Mozilla Public License
- NDI: Non-developmental item (see COTS)
- OMB: Office of Management & Budget
- OSDL: Open Source Development Labs
- OSI: Open Source Initiative (opensource.org)
- OSJTF: Open Systems Joint Task Force
- OSS: Open Source Software
- PD: Public Domain
- PM: Program Manager
- RFP: Request for Proposal
- RH: Red Hat, Inc.
- ROI: Return on Investment
- STIG: Security Technical Implementation Guide
- TCO: Total Cost of Ownership
- U.S.: United States
- USC: U.S. Code
- V&V: Verification & Validation

Trademarks belong to the trademark holder.



IDA

Released under CC BY 3.0 License

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YY)
July 2012

2. REPORT TYPE
Presentation

3. DATES COVERED (From – To)

4. TITLE AND SUBTITLE
Open Source Software (OSS or FLOSS) and the U.S. Department of Defense (DoD)

5a. CONTRACT NUMBER
DASW01-04-C-0003

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBERS

5d. PROJECT NUMBER

5e. TASK NUMBER
GT-5-3329

5f. WORK UNIT NUMBER

6. AUTHOR(S)
David A. Wheeler

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES
Institute for Defense Analyses
4850 Mark Center Drive
Alexandria, VA 22311-1882

8. PERFORMING ORGANIZATION REPORT NUMBER
IDA Nonstandard Document NS D-4677
Log no. H 12-001092

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Georgia Tech Research Institute
1700 North Moore Street, Suite 1910
Arlington, VA 22209

10. SPONSOR'S / MONITOR'S ACRONYM
GTRI

11. SPONSOR'S / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; unlimited distribution: 14 May 2012.

13. SUPPLEMENTARY NOTES
This presentation is released under the Creative Commons Attribution 3.0 Unported (CC BY 3.0) license.

14. ABSTRACT
This presentation provides an overview of open source software (OSS) in the U.S. Department of Defense (DoD). The presentation covers the history, why the government or DoD might use OSS, and explains DoD OSS policy. It explains that OSS is commercial software, and discusses common OSS licenses.

15. SUBJECT TERMS
Cloud Computing, Cyber Workforce

16. SECURITY CLASSIFICATION OF:
Unclassified

a. REPORT
Unclassified

b. ABSTRACT
Unclassified

c. THIS PAGE
Unclassified

17. LIMITATION OF ABSTRACT
Unlimited

18. NUMBER OF PAGES
34

19a. NAME OF RESPONSIBLE PERSON
David A. Wheeler

19b. TELEPHONE NUMBER (Include Area Code)
703-845-6662