A Spotter's Guide To AACS Keys

pde@eff.org

AACS

- Digital Restrictions system for HD-DVD & BluRay
- Incredibly elaborate:
  - content encryption
  - 4 revocation mechanisms
  - 3 watermarking schemes
  - multiple separated security models
- Well over a dozen kinds of keys!
- BluRay can also use BD+ (!!!)

Digg an AACS processing key

The best riddle you will hear today. Period.

1. Directory assistance service number in Moscow, Russia...?
2. A function key on a computer keyboard following F8 and preceding F10...?
3. Yao Ming's (Houston Rockets) jersey number...?
4. Scientific symbol for molecular oxygen also known as dioxygen...?
5. Number of New York state route also known as a Rear Mountain-Beacon Highway...?
6. Number of U.S. interstate that runs from Iowa to Ohio...?
7. Non immigrant visa allowing Australian citizens to live and work in the United States...?
8. Number of vertebrate animal form, required for all research involving vertebrate animals that is conducted in a regulated research institution...?
9. The eight-sided die is also known as...?
10. The international direct dialing code for Switzerland...?
11. The number of consecutive games in which Joe DiMaggio had a base hit in 1941 (still a record)...?
12. The fifth generation of the Chevrolet Corvette sports car is known as...?
13. NBA playoff record for most points scored in a playoff game, achieved by Michael Jordan in 1986, is...?
14. The number of men who signed the United States declaration of independence in 1776...?
15. Atomic number of radium...?
16. IATA code for Centralwings Airline...?

ANSWERS:

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.
09	**F9**	**11**	**02**	**9D**	**74**	**E3**	**5B**	**D8**	**41**	**56**	**c5**	**63**	**56**	**88**	**c0**

Talk Objectives

- Explain the system, so the community knows what to expect:
  - Subsystems
  - Keys
- Implications for free software?
- Reduce DOS against researchers
- Fight zombies

This talk (cont)

- Reporting on an ongoing effort to clearly explain all of the AACS architecture
- Mostly based on public sources
- The system is complex; we probably have a mistake or two somewhere!
- So, what is this beast?

Zombie DRM

It keeps comin' back

Interesting AACS subsystems

- Media encryption & receiver revocation
- Drive/host mutual authentication & revocation
- Video watermarking & traitor tracing
- “Theatrical” and “consumer” audio watermarking

More AACS subsystems

- Content revocation
- Managed copy
- Downloaded “extras”
- I'm sure I missed some...

Divide-and-conquer the users

- Two robustness models:
  - hardware tamper resistance
  - proactive renewal (mostly for software)
- Two classes of keying:
  - KCD (“key conversion data”) ~for hw players
  - Non-KCD ~for sw players
- Why, you ask?

Zombies split the party

- A class-break against hardware will not yield device keys that work with ordinary PC drives
- PC readers do not (or cannot physically?) read KCD
- KCD implementation details are secret

Some AACS keys (!)

DRM Subsystem(s)	**Keys**	**Other important data structures**
Subset-difference encryption & revocation	Device keys (including hw, sw, subsidiary and leaf device keys), processing keys, media keys, volume unique keys, title keys	Subset-difference records (in 2nd half of MKB)
Drive/host authentication & revocation	Drive public keys, host public keys; *not yet deployed*: bus session key	Host ID & certificates, drive ID & certificates, host revocation lists, drive revocation lists
Content revocation	(only AACS LA public keys)	Content certificates, content revocation lists
Video watermarking & “traitor tracing”	sequence keys, segment keys, media key variants, volume variant unique keys	Sequence Key Block

Media encryption & revocation

- AES encrypts the video stream
- Who can calculate the key?
- Any player who has not been revoked for this (or later) discs
- This is achieved using a *subset-difference key tree*

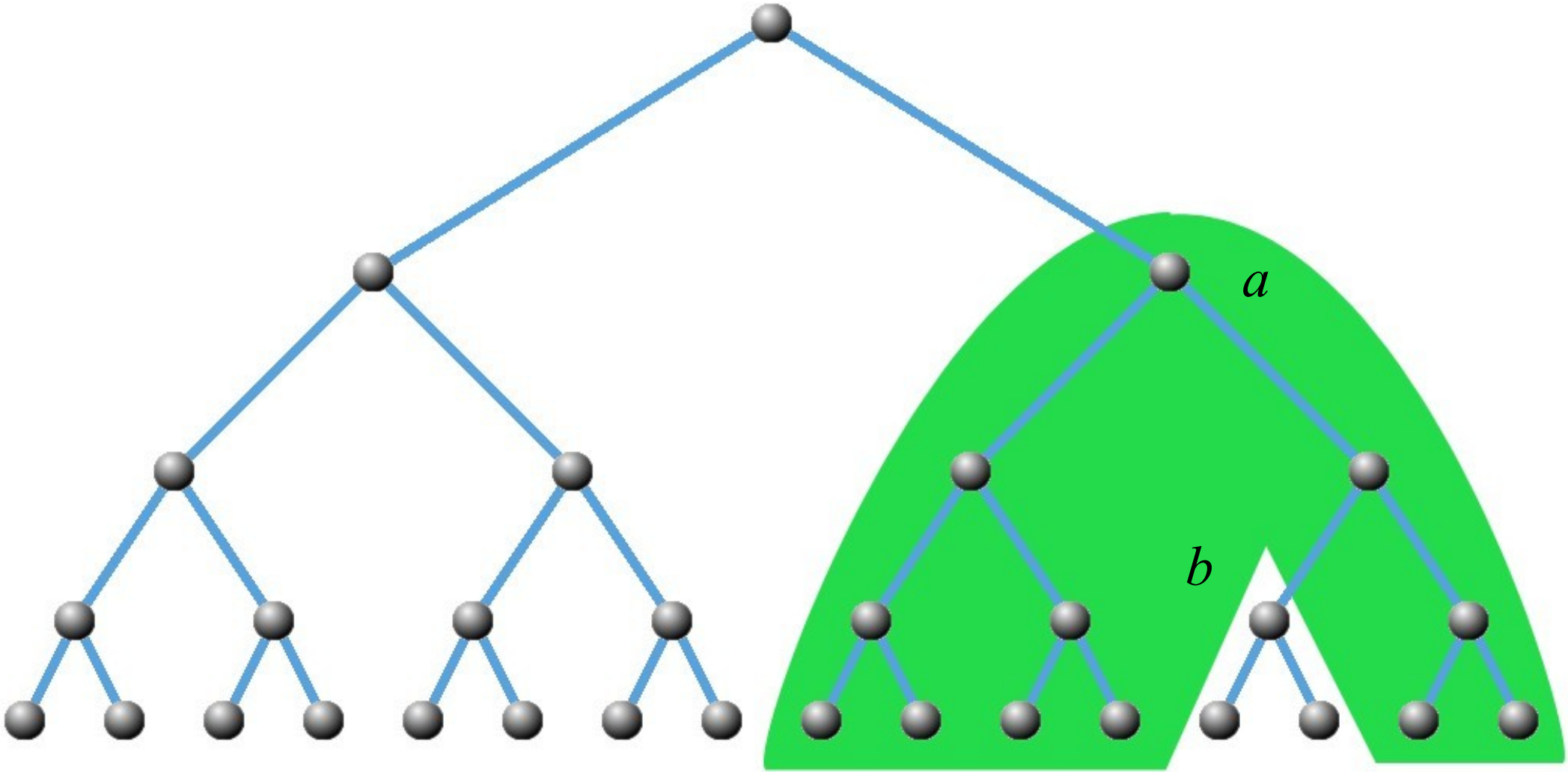
Subset-difference key tree

- Proposed by Naor, Naor & Lotspiech (2001)
- A large, virtual data structure
- Any subset of devices can be revoked
- Revocation by a new “Media Key Block” (MKB) header on new discs
- Each MKB uses a different bits of the subset-difference key tree

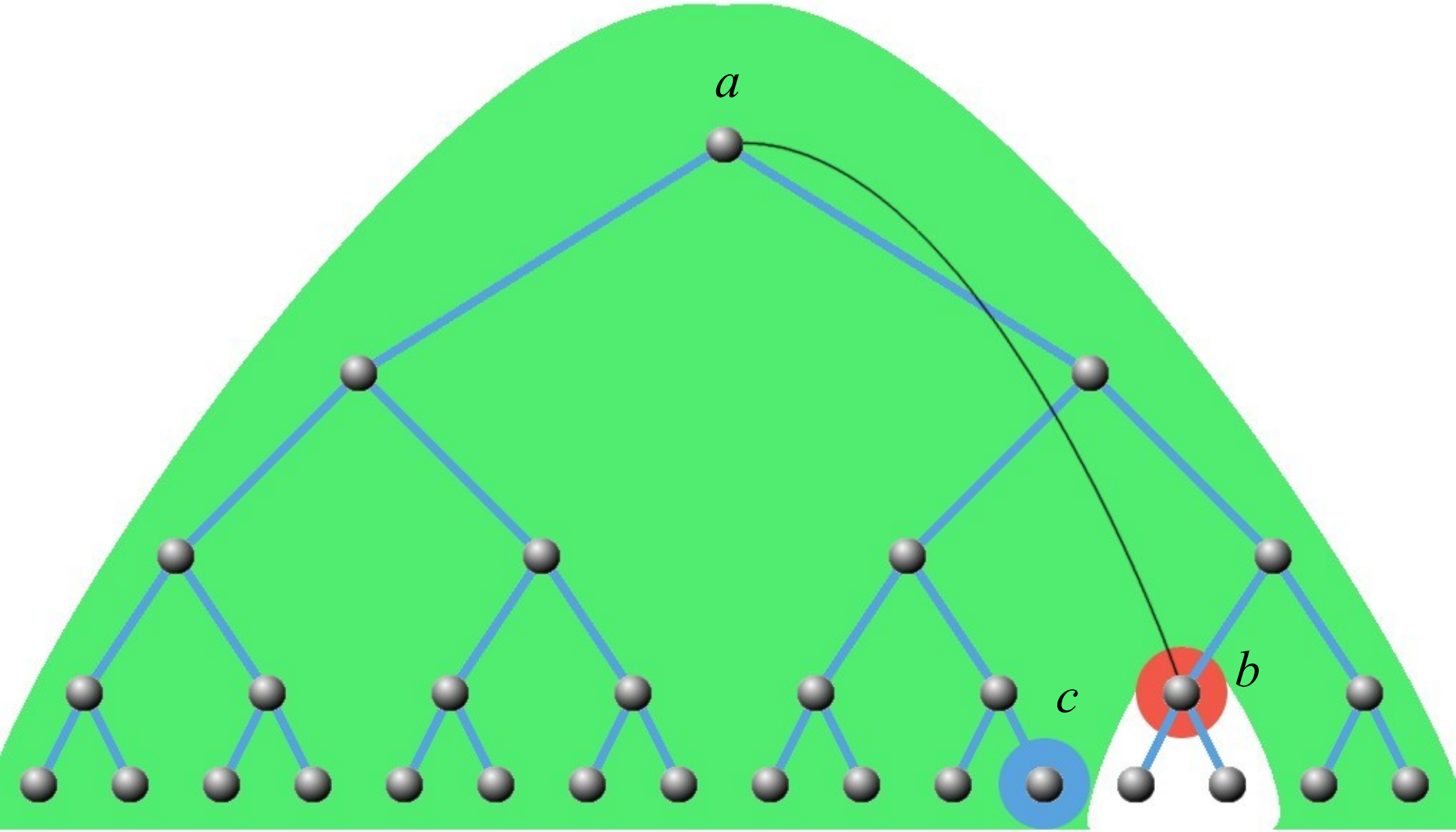
The subsets

- Devices are grouped into some *subsets*
- Every possible subset has a *processing key*
- A subset/processing key is made usable by encrypting a *media key* with it in the MKB
- If you are not in any of these subsets, you have been revoked

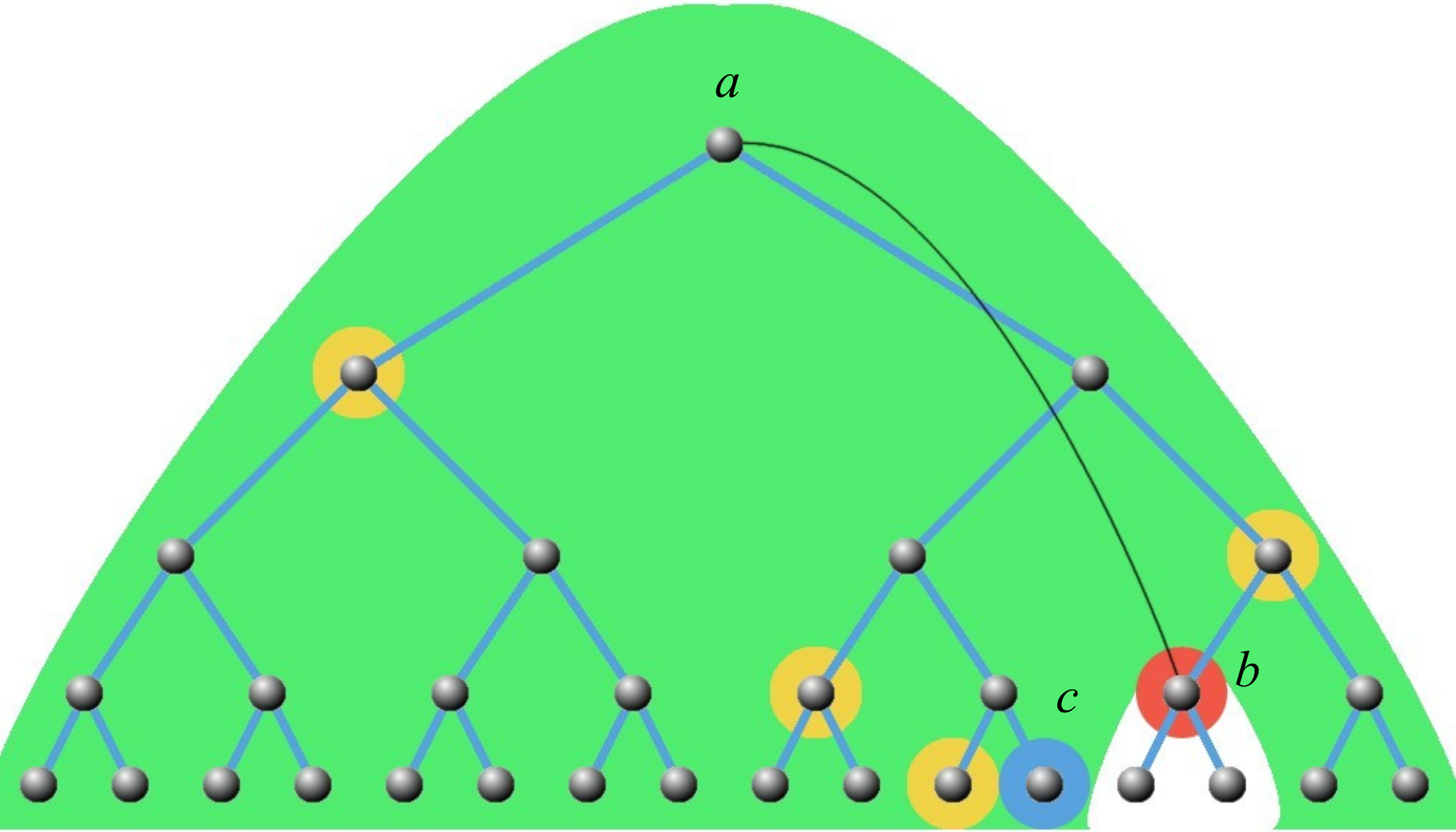
Subsets are of the form (subtree a – subsubtree b)



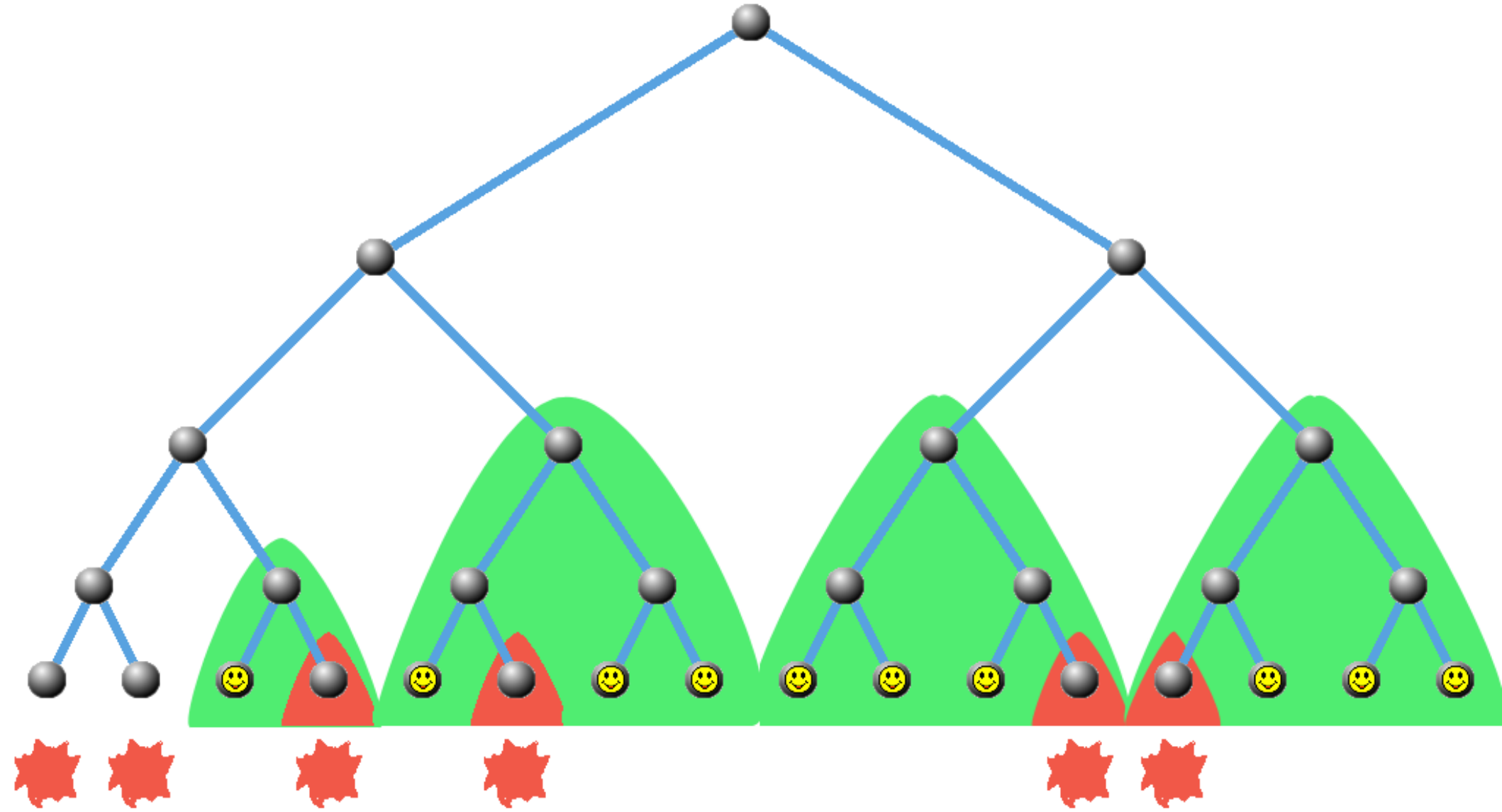
Each possible a induces its own tree of processing keys;
the one induced by a at b is the key for the subset
All the other leaves like c can calculate that key... how?



c can calculate any key here except its own, starting with the yellow *device keys*



How to encrypt for all but *r* users?



Subset-difference tree efficiency

- N total users, r of them revoked
- Up to $2r - 1$ entries in header, $1.25r$ on average
- Requires $\frac{1}{2}\log_2^2 N + \frac{1}{2}\log_2 N + 1$ keys / device
- But we don't know if AACS-LA is actually doing it this way...
- How big is N?
  - Lotspiech says 2^{31}
  - maybe 2^{22} for now?

Keys so far?

Key name	Unique to	Stored on / held by / issued to	Replaced / updated by	Revoked / invalidated by	Encrypts	Encrypted/ hidden with
device keys (hardware)	hardware players	253+ per player; more derived on the fly	N/A	new MKB or drive authentication	processing key	tamper resistance
device keys (software)	software player versions		software update	new MKB or host auth		software/OS measures
leaf key (a kind of device key)	players	$\log_2 N$ exist per device	N/A	new MKB	may encrypt your processing key, if you are revoked	every device in a subset but d can calculate d's leaf key
processing keys	1 per device key	calculated from device keys	as for device keys		see next row	same as device keys
usable[8] *processing keys*	MKB versions (1st half of MKB)	up to $2r-1$ per MKB,[9] $1.25r$ on average	new MKBs	on new discs	media key	subset-difference tree
media key	(MKB version, disc)?	in 2nd half of MKB entries	remastering of that title	remastering of that title	title key (via VUK), variant data	processing key, KCD (hw players)
volume unique key	$=f$(media key + volume ID)	calculated by player	remastering of that title	remastering of that title	title key	same as media key
title key	each title released	each disc	remastering[10]	remastering	video data	media/volume unique key

3 Watermarks

- Two not-yet-implemented SDMI-style audio watermarks....
- These will be coupled to refuse-to-play logic in devices:
  - “theatrical” watermarks that always abort playback
  - “consumer” watermarks that cause an abort after a certain amount of time
- SDMI-style watermarks seem to break
- And one much more serious watermark...

Video watermarking

- Based upon variant “marks” in the film
- Immune to transcoding & other noise
- May subtly visible:
  - a twinkle in an character's eye
  - a very slightly different camera angle
- Up to 30 binary marks (6144 variants) of a movie per disc
- What do the marks do?

“Traitor tracing”

- Uses an sophisticated scheme from Boneh and Shaw (1998) or somewhere in the subsequent literature
- Could theoretically trace hackers if their device purchases are identifiable
- Likely to be resistant against coalitions of some size c

“Traitor tracing” details

- Algorithm is secret
- Tradeoff between c and the number of marks required for tracing
  - Large c (>> 10?) may be prohibitively costly
  - But check the recent literature!
- 1024 is not many variants
- *Attackers can see when they have the set*
- Large coalitions *might* frame innocent users

Watermark playback

- Another whole set of keys and data structures
- Another complicated table

So, who fights zombies?

Pirates of the Caribbean

- Slysoft
  - based in Antigua
  - proprietary Windows binary blob
  - regularly updated
    - new host public keys
    - new device/processing keys
  - they're breaking MKBs faster than AACS-LA can update them

Prospects for sane AACS playback on free/open source platforms

- Not so good...
- Host authentication is a problem!
  - Someone obtains a Host Public Key every few months
  - Hacked or modchipped drives
- Also need a supply of device keys
  - (less time sensitive)
- GNU/Linux users will be driven to piracy

Some things that break AACS forever?

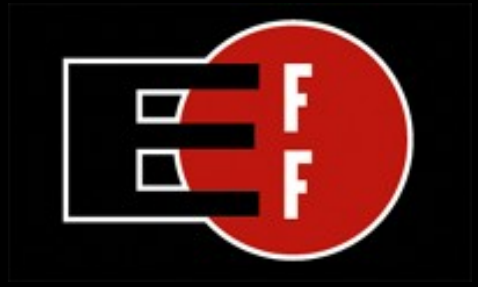
- Large databases of title keys or extracted KCD
- Any non-KCD hardware players + modchips could lead to a steady supply of non-KCD device keys
- Cryptanalysis or leaks of AACS-LA secrets (unlikely)

Legal Issues

- AACS implementations controlled by *hook IP*
- EFF has published some legal information
  - Google: *09 f9: a Legal Primer*
  - US / DMCA specific
  - Other WIPO Copyright Treaty implementations vary
- Research is risky
  - But this depends on national law

Conclusions?

- AACS is not going to stop large scale piracy
- *AACS is a powerful platform control tool*
- This will be a major pain for free software developers and users
- A major waste of some very intelligent people's time



About EFF

- Public interest tech policy organisation
- Mostly in San Francisco
  - We have a small office in Brussels, too
- Member funded
  - Please join, we'll send you a t-shirt :)
- www.eff.org