DRM

Digital Rights Management

What is DRM?

- New technologies bring with them new issues:
  - Advances in compression techniques make it possible to create high-quality digital content (audio, video, still pictures, etc.)
  - Advances in the network protocols and infrastructure makes it possible to store, stream and distribute this content in a very large scale.
- Digital Rights Management (DRM) is the set of techniques used to:
  - Control access to content:
    - Viewing rights
    - Reproduction (copying) rights
- Essentially, DRM is the management of the author's and publisher's intellectual property in the digital world.

DRM Principles

- Encryption of the content to disallow uncontrolled access.
- Decryption key management.
- Access control according to flexible usage rules
  - Number of times content can be accessed; times it can be accessed; trading of access rights.
- Copy control or copy prevention
  - Management of the number of copies that can be made of the content.
- Identification and tracing of multimedia data.
  - May be a requirement even if the copy is made from the analog version of the content, e.g., recording the analog outputs of a digital playback.

Underlying Technologies

- DRM is based on two fundamental underlying technologies:
  - Encryption
  - Digital Watermarking
- ***Encryption*** is used to "lock" the content and deny access to it to those parties that do not possess the appropriate keys
  - Encryption enforces the restrictions placed on the content by the author/publisher
- ***Digital Watermarking*** is used to "mark" the content so that a particular copy can be traced back to the original user
  - Digital Watermarking is used as a deterrent to large-scale unauthorized copying of copyrighted material.

Types of DRM

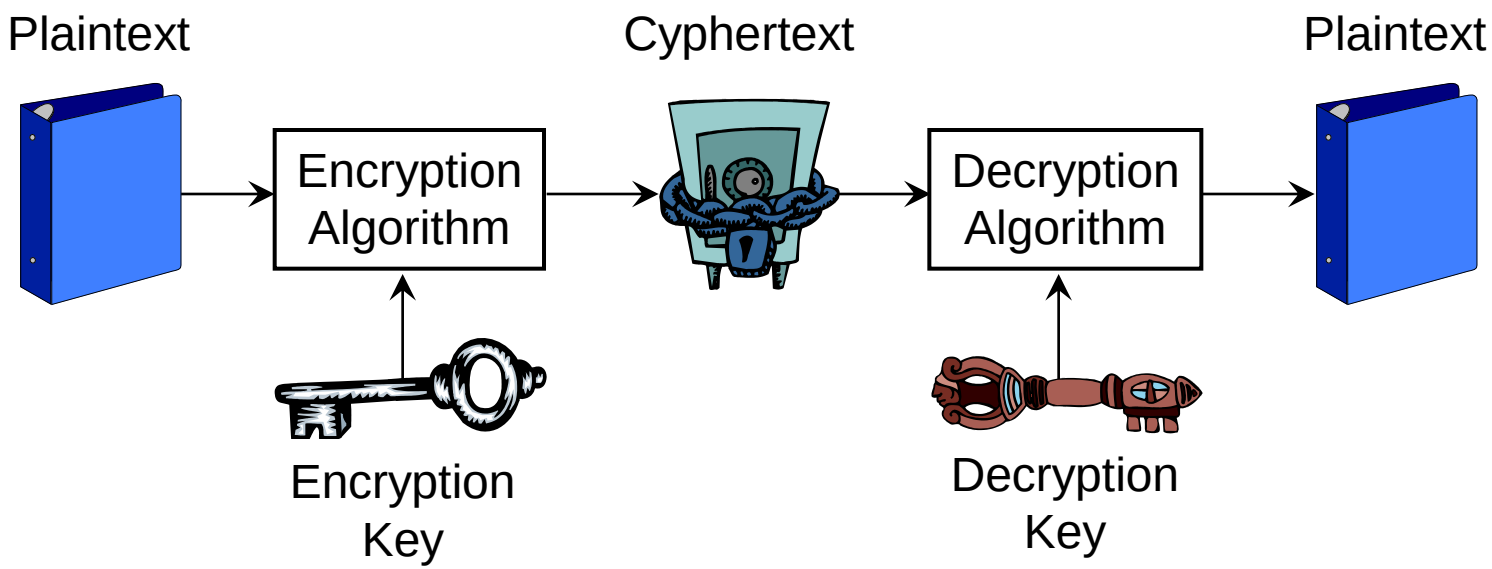
- "Transport" or "Link Layer": protects only the channel between the source and the recipients of the media.
  - DVB-style scrambling, used to protect MPEG-2 Transport Streams over satellite or digital cable.
  - IP Security.
- End-to-End: protects the object or media, and is embedded in it.
  - Independent of the transport.
  - No requirement of link-layer DRM.

Outline

- Technologies
  - Encryption Basics
  - Watermarking Basics
- Transport-Level DRM
  - DVB-style scrambling
  - IPSec
- End-to-end DRM

Encryption

- Encryption is the process of "obscuring" a message (content, media, file, etc.) so that it is undecipherable without the key.



Types of Encryption

- ***Symmetric Encryption***: the encryption and decryption keys are the same.
- ***Asymmetric Encryption***: keys come in pairs, one to encrypt and another to decrypt.
  - Used in Public-Key cryptography, where one key in the pair is kept secret, and another is published.
  - Whatever is encrypted with one key can only be decrypted with the other and vice-versa.
- Asymmetric Encryption is much slower than Symmetric Encryption and requires much larger key lengths to achieve the same level of protection.
- Some asymmetric encryption algorithms (RSA) have the limitation that the data size must be less than the key size.

Key Exchanges

- Symmetric keys are very efficient, but need to remain a secret and must be securely communicated between the participants.
  - Example: the NSA uses a fleet of airplanes and armed couriers to shuttle around 15 tons of paper containing symmetric keys per year for use by the US Government.
- Asymmetric keys (public/private) are slow and inappropriate for actual content exchange.
- Idea: use asymmetric keys to encrypt the symmetric keys, in order to securely communicate them.

The Diffie-Hellman Key Exchange

- Alice and Bob wish to agree on a secret key. They will use an open unsecured channel to negotiate that key.
- Start with a public base g and a public modulus p (everybody knows them).
- Alice chooses a large integer a and Bob chooses a large integer b.
- Alice computes $(g^a \bmod p) = A$ and sends A to Bob. Bob computes $(g^b \bmod p) = B$ and sends B to Alice.
- Alice uses $(B^a \bmod p)$ as her secret key; Bob uses $(A^b \bmod p)$ as his secret key; both are the same since they both are equivalent to $(g^{ab} \bmod p)$. An observer cannot easily figure out the key from g, p, A and B.

Diffie-Hellman Example

- Pre-selected: base $g = 2$, modulus $p = 9$.
- Alice chooses $a = 11$; $2^{11} = 2048$; $2048 \bmod 9 = 5$.
- Alice transmits $A = 5$ to Bob.
- Bob chooses $b = 8$; $2^8 = 256$; $256 \bmod 9 = 4$.
- Bob transmits $B = 4$ to Alice.
- Alice uses $4^{11} \bmod 9 = 4194304 \bmod 9 = 7$ as the shared secret.
- Bob uses $5^8 \bmod 9 = 390625 \bmod 9 = 7$ as the shared secret.
- See: `http://www.narf.com/~awestrop/crypt/dh.htm`

Watermarking

- Watermarking is the addition of unremovable data to multimedia content, for the purposes of copy identification and tracking.
- The requirements for such a system are:
  - ***Imperceptibility:*** the addition of the watermark must not degrade the content in a perceptible way.
  - ***Security:*** the watermark must only be accessible by authorized parties.
  - ***Robustness:*** the watermark must survive data manipulation, including malicious manipulation with the intent of removing the watermark.

Basic Idea

- Apply slight changes to the individual basic entities of the media.
- Since slight changes are vulnerable to manipulations and attack, spread them over a large portion of the host data.
- To prevent the watermark from being accessed by unauthorized parties:
  - Use some sort of secure cryptographic key.
  - Use watermarks that resemble noise; since media also has natural noise, watermarks “hide” in it.

Watermarking of Text

- Watermarking of formatted text is done using one of the following techniques:
  - Line shift coding: moving the lines of text up or down slightly; information is encoded in the way lines are shifted.
  - Word shift coding: same idea, but using spaces between words; much harder to extract.
  - Feature coding: slightly modify features such as the end line lengths of characters such as *b*, *d* and *h*.
- These techniques survive printing, consecutive photocopying up to 10 generations, and scanning.
- Easy to defeat, however: retype the text (OCR or manual).

Watermarking of Still Images

- There is a large body of techniques and literature on watermarking of still images.
- In general terms, the watermark is applied to the original, uncompressed image.
  - Some watermarks are designed in the space domain, while others are applied in the frequency domain.
- Some watermarks are designed to survive still image compression (JPEG), while others cannot.
- Simplest technique: replace the LSB of each pixel by a bit from the watermark.
  - Watermarks will be encoded in sequences of bits.
  - Image may be compressed to less bits prior to the injection of the watermark.

The Spread Spectrum Technique

- Steps to insert the watermark:
  - Select a pseudo-noise sequence, drawn from a random noise generator.
  - Modulate this sequence by the watermark data (use +1 and -1 as the data amplitudes).
  - Add the modulated sequence to the host signal (e.g., 1%of the luminance signal.
- Steps to retrieve the watermark:
  - Subtract the original image from the modified image to get at the noise.
  - Multiply the result (demodulation) by the same sequence used in the modulation and add the result over the samples that represent one bit of information.
  - If positive, watermark was +1; otherwise, it was -1.

Sample Images

Original

Watermarked

Source: **`http://dynamo.ecn.purdue.edu/~ace/water2/digwmk.html`**
Prof. Edward J. Delp's research group

Video Watermarking

- In general, the same techniques used for still images can be applied to video.
- Considerations:
  - The signal space for video is much larger than for still images; there is no need to use very complex schemes to minimize distortion while maximizing capacity.
  - Video watermarking schemes need to be less complex because in most cases they need to run in real time and need to address compressed video.
  - Video watermarks must be able to survive frame averaging, dropping and swapping - spread information over multiple frames
  - Depending on application, it is desirable to retrieve the watermark from short sequences from the material.

Video Watermarking Techniques

- DCT-based method:
  - Use the watermark to modulate a pseudo-noise signal of the same dimensions as the video.
  - Compute the DCT of the watermark and add it to the DCT of the original video.
  - Do not use the coefficient if this increases data rate too much.
  - Add drift compensation to avoid artifacts.
  - Typically capable of achieving around 50 bits/sec watermark.
- Motion-Vector method:
  - Find motion vectors that point to flat areas.
  - Slightly modify them to add the watermark information (randomized).
  - Watermark can be derived directly from motion vectors.

Audio Watermarking

- When compared with video, audio introduces the following issues:
  - Much less samples resulting in lower watermark capacity.
  - Humans are much less tolerant to audio changes than to video changes; harder to achieve imperceptibility.
- Basic spread-spectrum technique is also used for audio, but needs to be refined.
  - Example: making the power of the watermark signal vary with the overall power of the audio.
- Lot of activity in this area
  - See the "Secure Digital Music Initiative" (SDMI), at **`http://www.sdmi.org`**.

Transport-Level DRM: DVB Scrambling

- Issue: satellite TV, which uses MPEG-2 compression, is broadcast over the air.
- The broadcaster has the following requirements:
  - The signal can be received by anyone; however, unauthorized receivers must not be able to decrypt it.
  - If an authorized user stops paying, s/he must stop receiving the signal.
  - Multiple levels of authorization: some users are authorized to view more channels than others.
  - In general, user devices will not have a communications return path to the broadcaster → one-way communication using the MPEG-2 stream only!

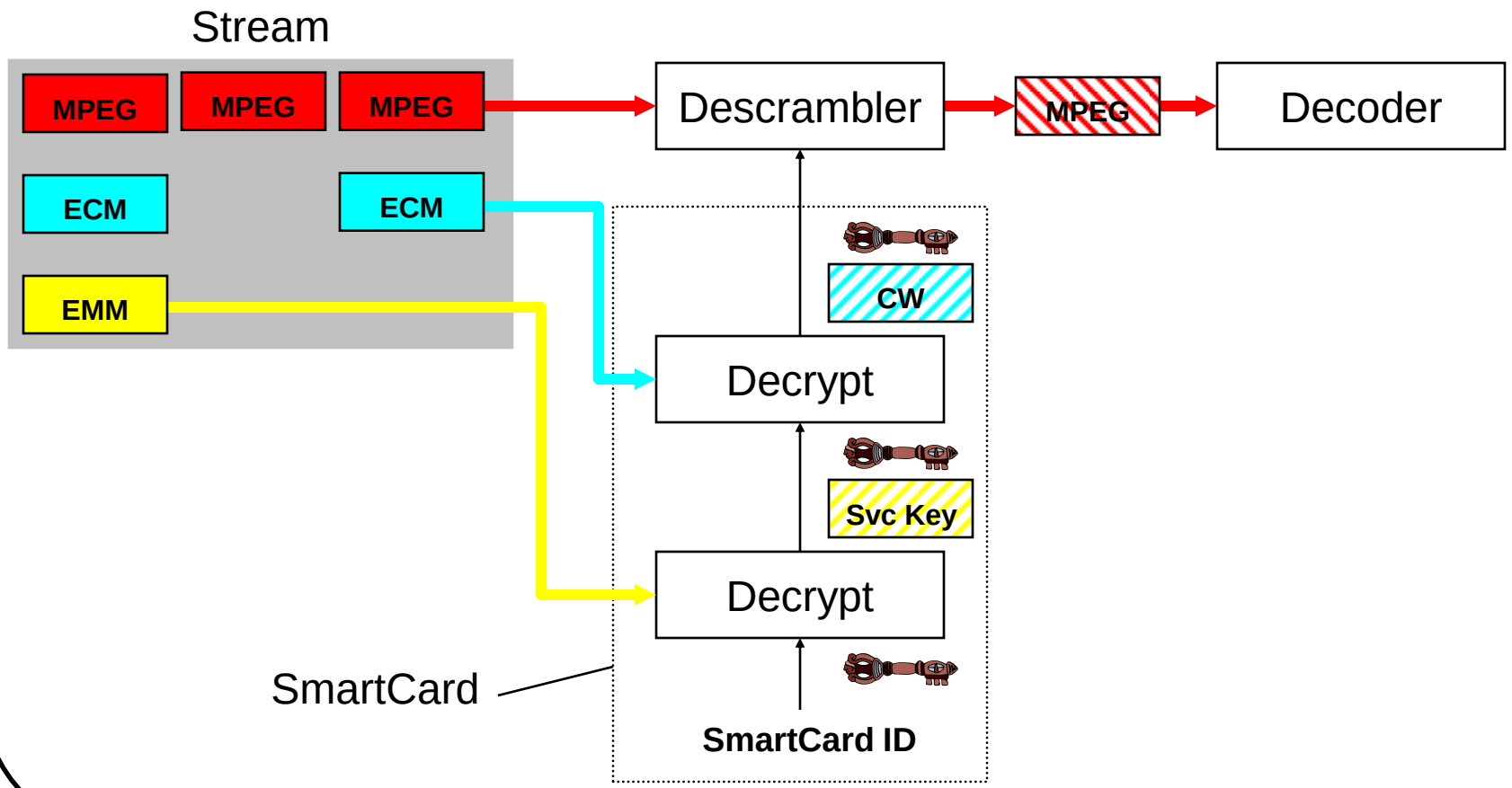
DVB Scrambling Basics

- DVB (Digital Video Broadcasting) systems use MPEG-2 Transport streams (fixed-size 188-byte packets, 4-byte header).
- Encryption applies only to the 184-byte payload; a flag in the header indicates whether the payload is encrypted.
- DVB calls the encryption of the payload "*scrambling*".
- The algorithm used in scrambling is called "*DVB Common Scrambling Algorithm*"; it uses a 64-bit key - the "*control word*", and the technical details of the algorithm are secret and only disclosed to hardware manufacturers.
- For any given stream, the control word is periodically changed. The time period a stream uses the same control word is called a "*cryptoperiod*".
- Cryptoperiods are typically on the order of 5 seconds.

DVB Scrambling Basics (cont.)

- The control words are embedded in the stream. They are carried in "*Entitlement Control Messages*" (ECMs). ECMs are encrypted using custom algorithms, proprietary to the Conditional Access vendor. The key used to encrypt the ECM is called the *Service Key*.
- DVB receivers normally have a *Smartcard,* provided by the Conditional Access vendor. The receiver will extract the ECMs from the stream and route them to the smartcard; if the user is entitled to see the program, the smartcard will decrypt the ECMs and return the control word.
- The service keys are carried in "*Entitlement Management Messages*" (EMMs), also embedded in the stream. The EMM is intended for an individual smartcard and is encrypted with that smartcard's individual key.

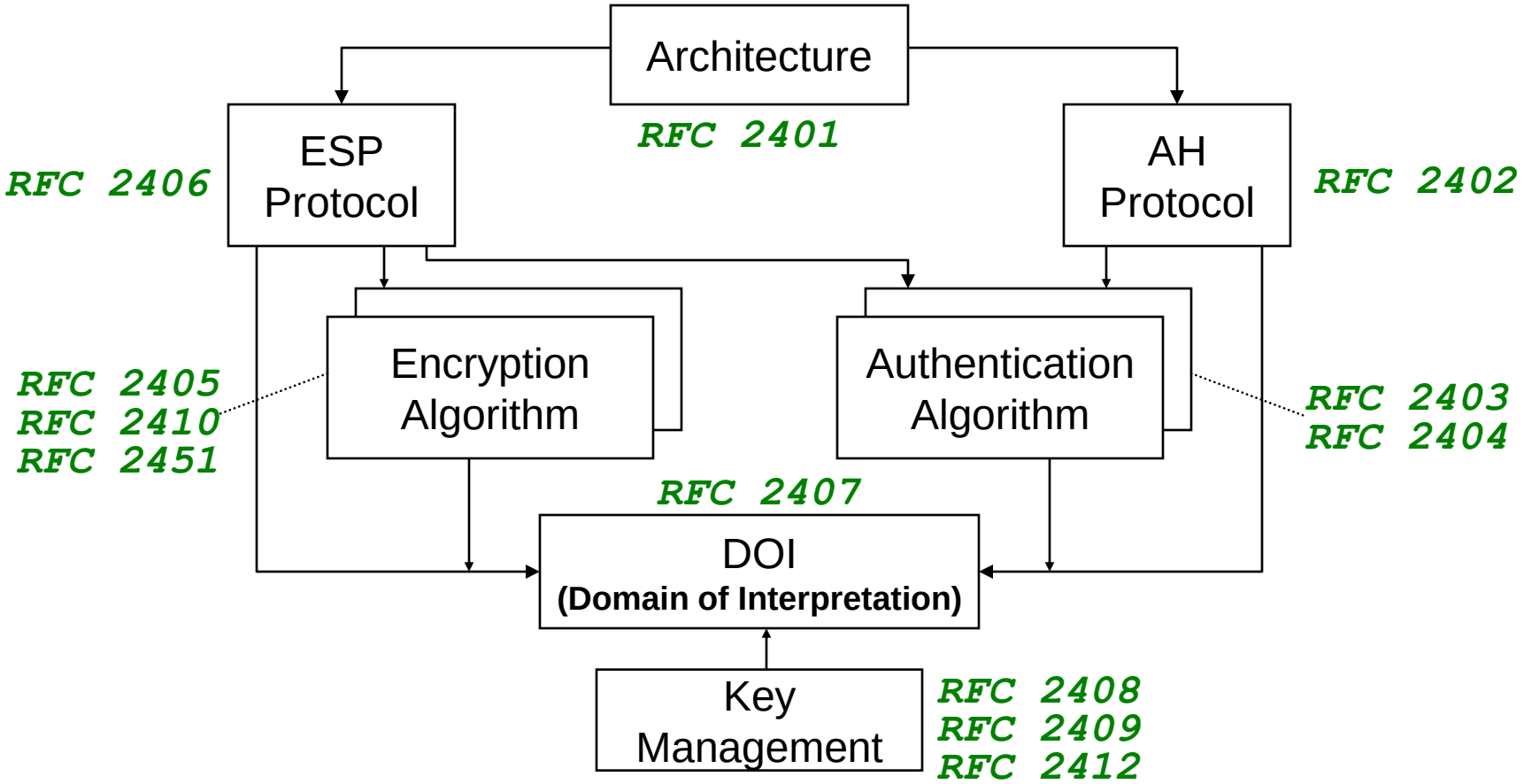
DVB Scrambling Illustration



Transport Level DRM: IPSec

- IP Security (IPSec) is the Internet standard for authentication and encryption of IP datagrams.
- There is also an Internet standard for lossless IP datagram compression (IPComp).
  - Compression and Encryption normally go together; if you are going to compress and encrypt, you need to compress first.
- The IPSec architecture, protocols and algorithms is defined by 12 RFCs (from RFC 2401 to RFC 2412, November 1998):
  - RFC 2401 is an overview of the architecture
  - RFC 2411 is a documentation roadmap
  - The other RFCs define protocols and algorithms
- IPComp is defined by RFCs 2393, 2394 and 2395.

IPSec Documentation Roadmap



IPSec: What it Does

- IPSec is deployed on hosts or on security gateways (intermediate systems implementing IPSec, such as a router or firewall).
- IPSec includes the following functionality:
  - selection of required security protocols
  - algorithm selection
  - key management
  - optional negotiation of IP compression
- IPSec provides the following security services:
  - access control
  - connectionless integrity
  - data origin authentication
  - rejection of replayed packets (a form of partial sequence integrity)
  - confidentiality (encryption)
  - limited traffic flow (source, destination) confidentiality

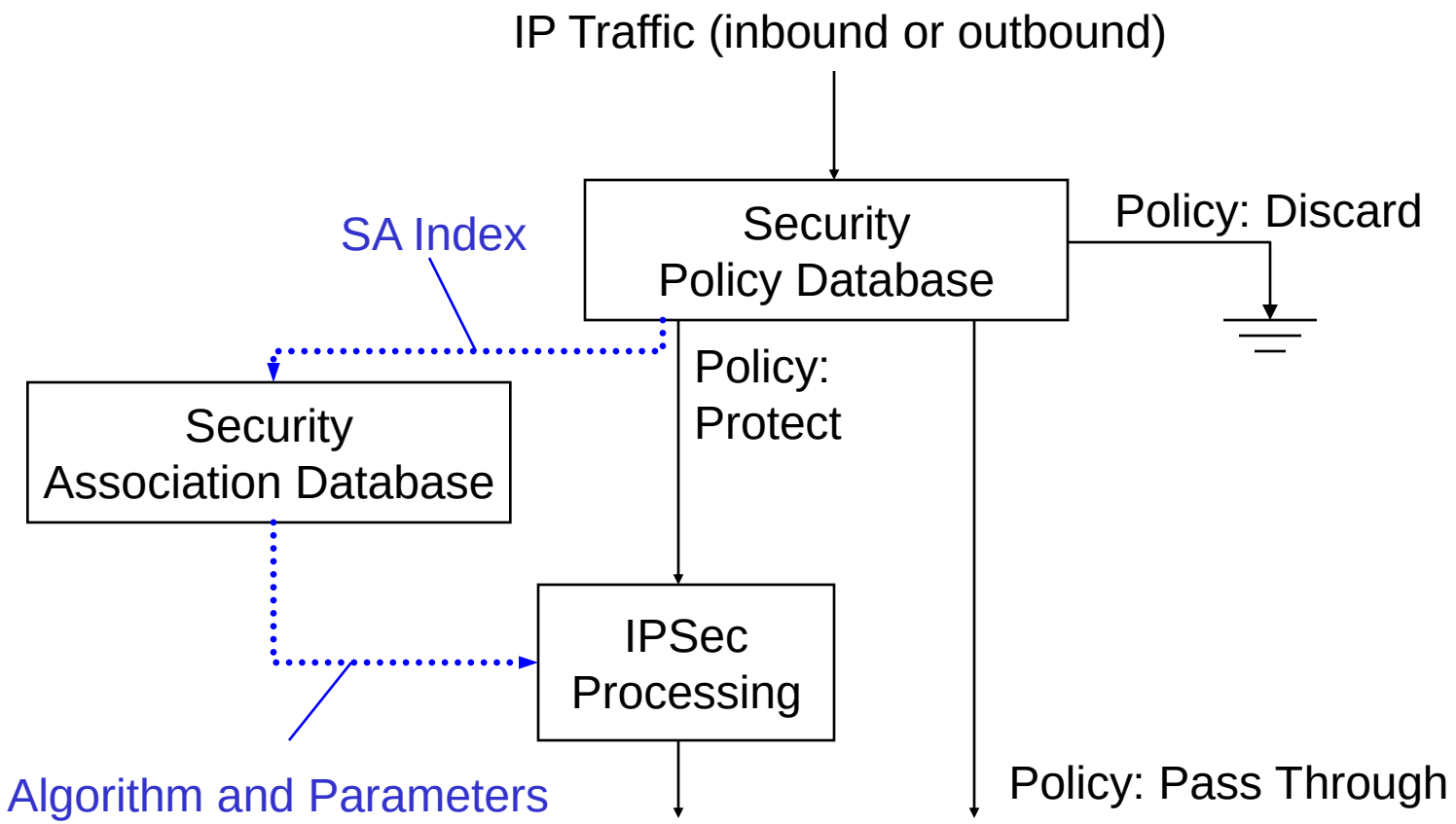
IPSec Basics

- IPSec can operate in the following two modes:
  - *Transport Mode*: end-to-end protection between two hosts. The IPSec header follows the IP header; protection is afforded only to the packet payload (some parts of the header may be afforded authentication).
  - *Tunnel Mode*: protection between two security gateways. The complete IP packet is encrypted and transmitted as the payload of an outer IP packet from one gateway to another.
- Since IPSec works at the datagram layer, it can be used to protect any upper level protocol (UDP, TCP, BGP, etc.).
- The actual encryption/authentication algorithms are negotiated by the endpoints and identified by index in the packet.
  - There are a number of standardized algorithms, but anybody can define a new one and use it with IPSec.

IPSec Databases

- A Security Association (SA) is a simplex "connection" that affords security services to the traffic carried by it.
  - If bi-directional security is required, then two SAs need to be created, one on each direction.
  - An SA only provides confidentiality or authentication; if both are needed, then multiple SAs need to be created.
- Conceptually, IPSec defines the following databases:
  - *Security Policy Database (SPD)*: specifies the policies that determine the "fate" of all inbound and outbound traffic for an IPSec-enabled host.
  - *Security Association Database (SAD)*: contains the parameters associated with each active SA.

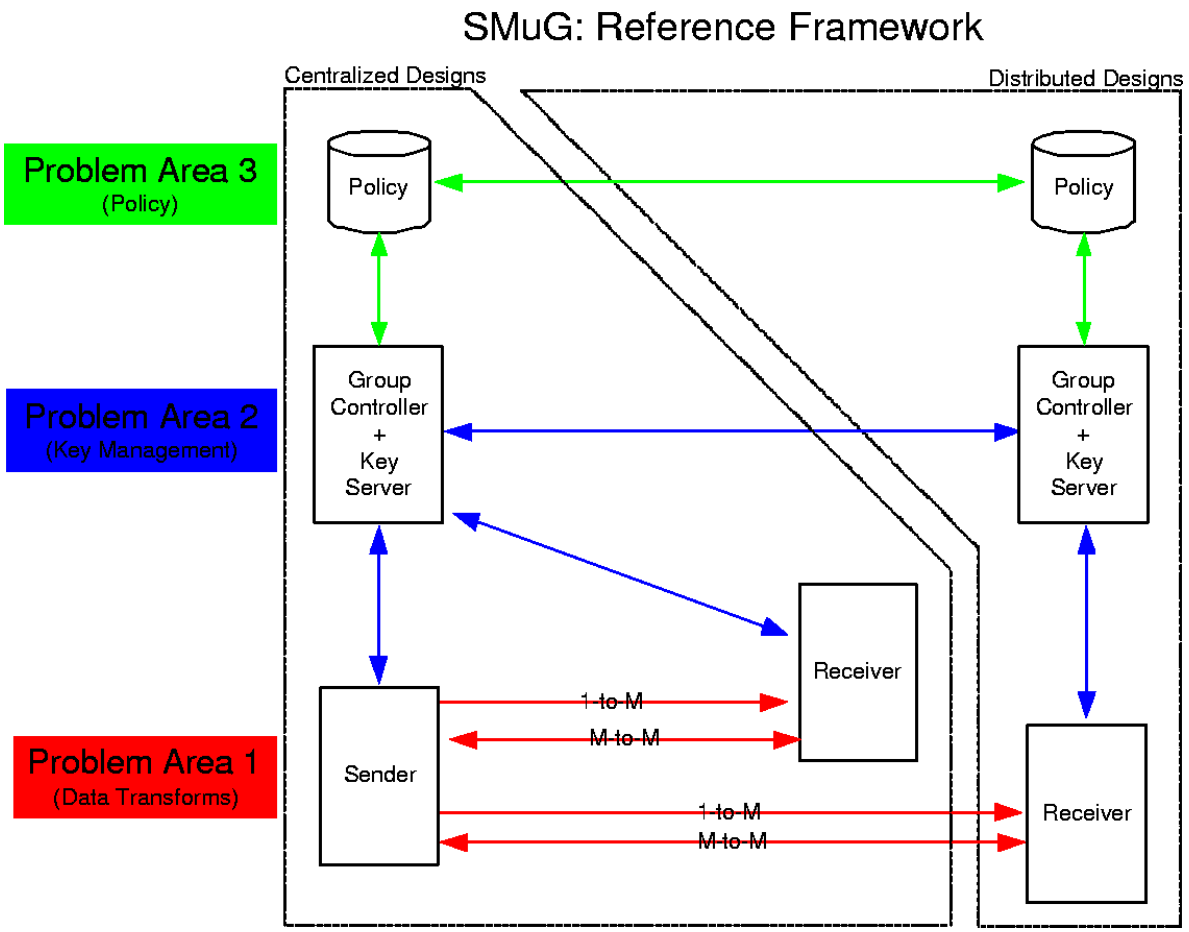
Processing Flow



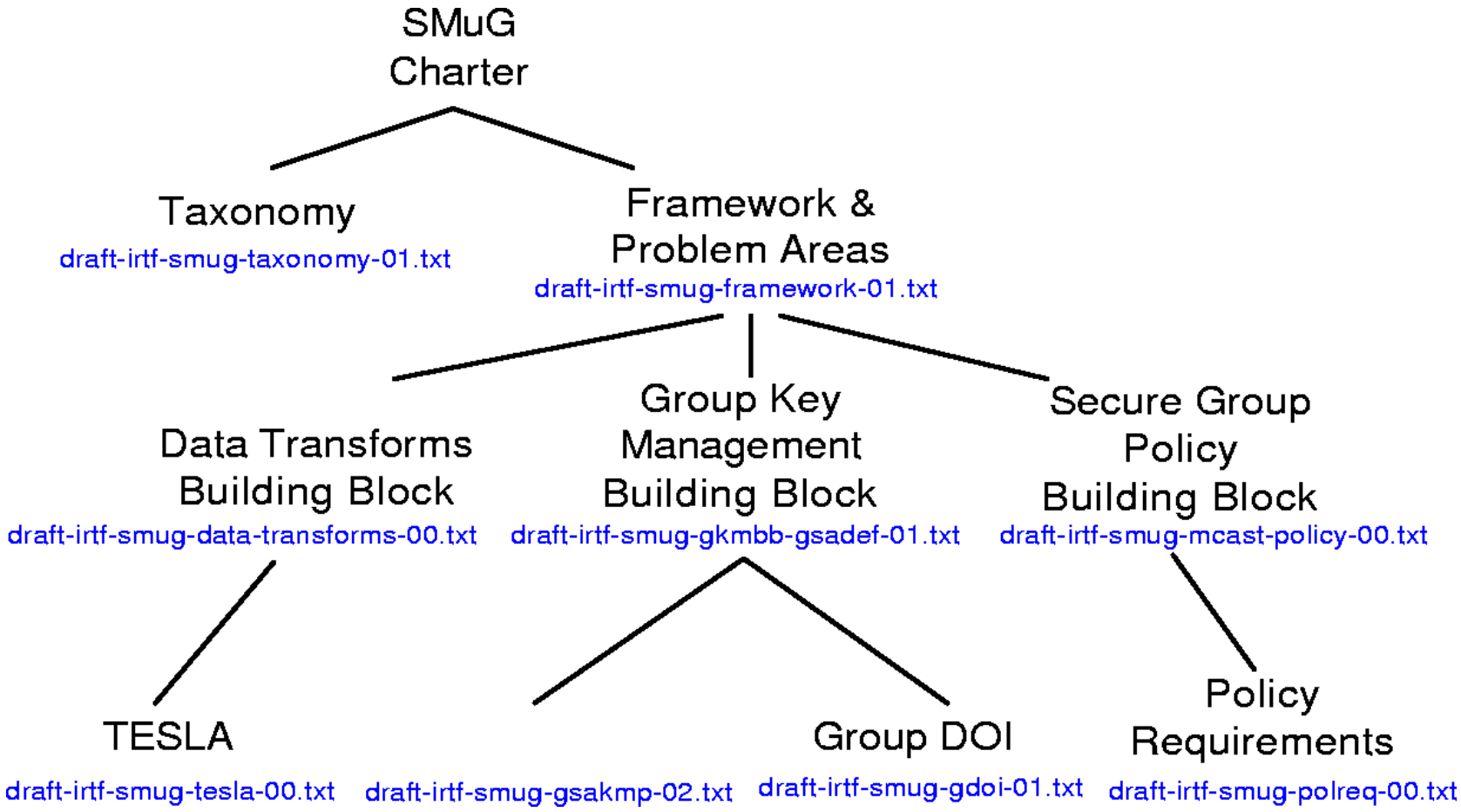
Securing Multicast

- One of the primary requirements of multimedia communication is multicast - and, by extension, security needs to be applied to multicast as well.
- There is considerable interest today in secure multicast as a means of distribution over the Internet.
- Although IPSec does not preclude the use of multicast, the key distribution issues are still very open - key negotiation, as currently defined, is a unicast operation.
- Secure Multicast Working Groups:
  - IETF Working Group: MSEC (Multicast Security), standardizing building blocks and protocols.
  - IRTF Working Group: SMuG (Secure Multicast Research Group), looking at defining a reference framework and building blocks.

SMuG Framework



SMuG: Draft Roadmap



DVB and Secure Multicast

- The DVB standard has solved the secure multicast problem very well for a specific environment: one-way broadcast of content to large number of users without a return path.
- Many of the DVB Conditional Access companies (Irdeto, Philips) have leveraged their technology into the IP Multicast security area.
- The same DVB paradigm is used, with ECMs and EMMs; the only difference is that these become IP multicast packets instead of MPEG transport packets.
- Many implementations require a SmartCard connected to the PC; serial and USB readers are available.

Issues with Transport-Level DRM

- Transport-level DRM is designed to protect the content in transit through an open network where it could otherwise be stolen.
- However, once content "emerges" on the other side of the transport link, it is unprotected.
- This may be acceptable in some cases, but other applications and newer business models require a more advanced level of protection, which needs to be end-to-end. For example:
  - Limiting the number of times a certain content can be copied.
  - Limiting the number of times a certain content can be played.
  - Limiting the number of times a certain content can be transferred.
  - Etc.
- Solution: end-to-end DRM, where the access rights become an integral part of the content.

End-to-End DRM Basics

- Content is packaged in a "container" where the access rights are specified. Encryption is used to enforce the access rights.
- The container can (typically) be freely downloaded or passed around.
- To open the container and access the media, the user needs a license, which can come from a number of places:
  - A commercial transaction with a web site.
  - Contact with a local SmartCard connected to a reader in the user's computer.
  - A file in the computer.
  - Identification of the hardware device where the content is being opened (e.g., players)

State of End-to-End DRM Today

- Several companies are playing in this space and offering toolkits and platforms for DRM:
  - Intertrust, Microsoft, IBM, ContentGuard, RPK, Sony, Reciprocal, …
- Microsoft has a (mostly) proprietary toolkit, which they give away for free.
- In the Digital Music area, companies got together and established the Secure Digital Music Initiative, a framework for compatibility.
- ContentGuard (a spin-off from Xerox, also backed by Microsoft), has put together XrML, an extension of XML to describe digital rights; the specification can be accessed for free.
- There has been a lot of activity in SmartCards, SmartCart readers, and Java-based SmartCards for user entitlement.

Suggested Reading

- General overview of DRM and watermarking:

  F. Hartung and F. Ramme, "Digital Rights Management and Watermarking of Multimedia Content for M-Commerce Applications," *IEEE Communications Magazine*, Nov. 2000, pp. 78-84.

- Detailed paper on watermarking:

  F. Hartung and M. Kutter, "Multimedia Watermarking Techniques," *Proceedings of the IEEE*, vol. 87, no. 7, July 1999, pp. 1079-1107.

- Light reading on DRM: white papers by Reciprocal.

  `http://www.reciprocal.com/abo_drm_knowledge.asp`

Suggested Reading (cont.)

- Software to attack and remove watermarks:

http://www.cl.cam.ac.uk/~fapp2/watermarking/image_watermarking/stirmark/

- Microsoft DRM Toolkit: Widows Media Rights Manager.

  http://www.microsoft.com/windows/windowsmedia/en/wm7/drm.asp

  (also check MSDN for Digital Rights Management APIs)

- Good overview of Public Key Cryptography:

  W. Diffie, "The First Ten Years of Public-Key Cryptography," *Proceedings of the IEEE*, vol. 76, no. 5, May 1988, pp. 560-77.

- Explanation of Diffie-Hellman without the math:

  http://securityportal.com/topnews/dhkeyexchange20000706.html

Suggested Reading (cont.)

- Cryptography basics
  - Tutorial:
    http://www.conceptlabs.co.uk/extras/IntroToCrypto-005a.zip
  - The Story of Alice and Bob:
    http://www.conceptlabs.co.uk/extras/ZurichSeminarSpeech3.htm
- DVB (Digital Video Broadcasting):
  - **http://www.dvb.org**
  - **http://www.coolstf.com/mpeg/**
  - **http://www.hf-fak.uib.no/smi/ksv/digfaq.html**
- Free IPSec implementation for Linux; check out their documentation pages, they have a good IPSec tutorial.
  - **http://www.freeswan.org/**
- Read RFC 2401, sections 1 through 4, for an overview of the IPSec architecture.

Suggested Reading (cont.)

- Secure Multicast Working Groups: charter and papers.
  - MSEC: **http://www.securemulticast.org/msec-index.htm**
  - SMuG: **http://www.securemulticast.org/smug-index.htm**
- Secure multicast overview, by the SMuG chair:
  - **http://www.stardust.com/multicast/whitepapers/security.htm**
- Very good overview of Secure Multicast by NDS (also includes a pitch for their products…)
  - **http://www.securegear.com/white_papers/nur111c.pdf**
- Secure Multicast Brochure by Irdeto Access:
  - **http://www.irdetoaccess.com/brochures/intro_multicast.pdf**
- XrML specification: free, but requires registration.
  - **http://www.xrml.org**

Business and "Social" Aspects

- Position Papers from the W3C Workshop on DRM:
  - **http://www.w3.org/2000/12/drm-ws/pp/Overview.html**
- Secure Digital Music Initiative:
  - **http://www.sdmi.org**
- What happened when the SDMI Watermarks got broken (after they asked for it):
  - **http://www.cs.princeton.edu/sip/sdmi/index.html**
- John Gilmore's position on this whole thing:
  - **http://lists.w3.org/Archives/Public/www-drm/2001Jan/0009.html**
- Papers by Andrew Odlyzko (read "Content is not King"):
  - **http://www.research.att.com/~amo/doc/eworld.html**