Actual Cryptography at the Age of Evolving Ecosystems

Moti Yung,

Google
Talk Agenda

• Part I: Crypto as part of general engineering projects
• Part II: Adx—Review
• Part III: Adx—Crypto solutions
• Part IV: Conclusions
From Abstract to Actual Crypto

• **Abstract**: Cryptographers deal with models, nicely quantified “adversarial power,” then definitions, constructions, proofs, complexity,….

• **Applied**: looks at systems context and either applies a model to a sub-problem (authenticated key exchange, fast software encryption) and uses implementable primitives...

• **Applied security system**: natural; creating building blocks/systems/protocols/standards: EAS, RSA, TLS, SHA..

• **Actual crypto eng.**: deploy specialized or novel custom made crypto in general system within actual development and deployed systems.
My Goals in this talk

• Actual crypto is different from abstract crypto since it is working in an actual systems context: development, maintenance, business.

• Try to reflect upon these questions:
  – How to take part in a global ecosystem development process (& its specialized crypto needs)?
  – How to make sure crypto extends and survives as the systems evolve?
  – The differences of aesthetics/ measures of achievement
Actual Crypto does not live alone

• Security is often at odds with, e.g.:
  – System Function
  – Performance
  – Usability (the User factor)

• Crypto is best applied when the above conflicts do not apply (e.g., hidden from the end user), or when the security requirement dominates (absolutely or to a large degree) and crypto aid security. (→ There is incentive to use crypto...)
  – This is an industrial perspective which is not in the textbook on crypto
The Economics of Development

• Computers and systems are designed to “compute a task” not to “be secure,” so we need to optimize the deployment of crypto; and this is an art (it may be formalized and cases have been analyzed: economics of secure systems: where the incentive lies?)

• → Security is a fundamental issue (needed/ hard), but of secondary importance (tolerated/ be cheap)

• → Security cannot be retrofitted, but it always is!! Since non-experts do not see a need....

• → Crypto/ security eng. has to be (positively) opportunistic!
Examples: crypto missed

- Database: not encrypted since relational algebra is hard on encrypted data.... (crypto goes against functionality and against performance)
- Early “secure mail” hard to configure so users chose the “insecure mode” as a default
- All routers same password: scalability of maintenance comes first, neglecting “real” security
- IBM’s SNA: password on the clear! Rely on physical security, when “network scaled across same branch” problem ignored!
- Protocol extended: security not reviewed!
Thus: we see

• After attacks which reduce the system’s availability to users, hurt performance and function, people will tend to invest more in security (incentives)
• Mission critical system: security is part of function
• The need for crypto may come from different sources, may be implicit in the spec, so need to look for where it applies first..(the path of least resistance). \(\rightarrow\) Need to be involved early!...and think carefully:
  – What is possible under the constraints?
  – Where and how to use the opportunity in the overall product context? Identifying initial well recognized need is important!
To Ad Exchange (ADX)
Internet Ads: Sponsored Search
Internet Ads: Display Ads
Internet/ Mobile Ads: Display Ads

• Traditional Online publishers and advertisers work together:
  – Negotiate offline or via intermediate networks,
  – Use planning, static policy, pricing and ad serving systems
    • DoubleClick, Microsoft’s aQuantive, AOL’s ADTECH AG, WPP’s 24/7 Real Media.
  – Efficiency, effectiveness of this bulky “brand advertiser” model?

• The Newest Proposal for display ad business:
  – Two-sided real-time marketplace for matching online publishers and (“direct response”) advertisers.
  – Yahoo’s RightMedia, Google’s Ad Exchange, Microsoft’s AdECN.

• It applies to web and mobile advertisements
Exchange

• On one side there are publishers (web pages) that have space for putting ads
• On the other side there are Ad Networks (buyers) representing companies that want to advertize. There are a few hundreds of those
• Ads are “added” to web pages

• There are many “viewers” of pages at publishers: every one browsing (essentially). Thus, this is a very huge scale Internet wide application
Advertizing

• Can be done by and via Ad networks directly (buying an ad)

• Can be done via the exchange/mixed models

• Let us review the Adx which resides in Google Cloud
AdX Model

Publisher → Ad X → Ad Network 1 → Ad Network 2 → Ad Network J → Advertiser 1 → Advertiser 2 → Advertiser n_j

viewer v, price p, Inv I → adx(v, I) → ad*, price q → auction → ad, bid b
Architecture:

- Viewer: you!
- Publisher: www.cnn.com
- AdX: the exchange hosted in Google Infrastructure available globally
- Ads Agency: Ads producer for companies (Coca Cola ads to be inserted) and distributors
- Advertiser: Coca Cola.
Evolution of AdX

• Doubleclick: modify to an exchange..
• Paper design, one server, three....
• Now: billions of transactions/ day, global exchange...
• The ecosystem of display is changing: mobile, apps, and so on...
To Security & Privacy
Immediate security

• The first goal in security was systems oriented: secure the user interfaces/ web/ ads/ anti-malware...

• ...and then we thought to crypto-secure the bids when needed since others should not learn them (according to the contract)...

  – Where are the possible leaks?

Then, we reviewed business and design and looked at added needs where is security/ crypto/ related issues needed?
AdX characteristics

• **Speed:**
  – Everything has to be done FAST (cannot slow down the Internet !!!).

• **Volume/ Scale:**
  – For a few years AdX runs ~**billions auctions /day** with a few (~thousands) networks.
  – High bandwidth requirement

• **Evolution:** design system for evolving “market place” & added requirements
Interesting Issue: After the Auction

• “Viewer’s page” redirected to Ad Network with “I frame for display” that has the winning price embedded in it (winning price macro) pull model

• Viewer gets the ad, winning price exposed to user (violates business agreement (contract) and practical engineering of exchange) → ??? “a problem”
  – Note it is not “on the wire” but at the browser!

• This is a call for action: an immediate issue needs solution, and an opportunity to introduce cryptography!
Security & Performance & Cost Align

- Embedded price in the macro (I frame) at the user possession that is used to pull the ad (for optimization need to send the price)
- This macro is the only way for the agency to know the price (second price auction; communication piggybacked).
- Otherwise: Hard to connect the price in another way to the agency (even if can double the bandwidth to the agency).
- Best way to send via the user the price (in fact, security is secondary to the need to employ the user as a channel). Thus: Security and Performance/ cost align together!!!!
- Gap between Business model (service agreement) and Engineering needs \( \rightarrow \) crypto to the rescue!!
Needed

• Secure delivery
• → analyze what encryption can be used (performance, context dependencies, security needs)
• → key management support
Crypto Designer Goals

• Have a general encryption utility for current and FUTURE security needs. Cannot utilize standard solutions (SSL...)—be opportunistic!

• Separate **key management**: generation, distribution, rotation (which can exploit existing components) and customized on-line operations.

• Provide a solution for **secrecy** and **integrity**.

• Volume implies: many times over the same cleartext values (**same price again and again**). Need to retain (semantic) security nevertheless→ **special security needs**
Crypto Designer Goals cont.

• Stay in touch with engineering team....since needs will surely come, and the tools/ hooks are already in the system!
Key management

• Auctioneer (Adx) and Ad Agency will exchange keys externally
  – Use out of band methods..
  – Or: use TLS/SSL relies on public key technology and on key exchange protocol (Signed Diffie-Hellman key exchange)
  – Typical solution: use the exchanged key. Can employ TLS w/ both sides having a public-key (server side and client side keys)
  – Result: both parties share a key for symmetric key use
Side remark: The guts of TLS/SSL

- $A \rightarrow B$: $g^A$ signed by public key of A
- $B \rightarrow A$: $g^B$ signed by public key of B

- $(g^A)^B = (g^B)^A = g^{(A*B)}$ is joint key from which to derive the key.

- This is just standard protocol but 1000 agencies and a single auctioneer can do it at no problem! Offline...
- Industry – you exploit existing solutions
Security in Operation

• The encrypted price goes via the user browser to the agency, user can learn & modify!
• Need to make sure the encryption is valid (unless user erases/spoils the encryption, in which case the agency knows not to take it into account → need to detect manipulations).
• The encryption has to be authenticated as original
Authenticated Encryption

• Combines Encryption and Authentication of the Encryption
  – Privacy: provides good hiding of the message
  – Authenticity: assures receiver that it comes from the original party
  – → any attempt to forge will fail with very high probability

• Around 2000 it began to be an area of research
Authenticated Encryption

• Preneel van Oorschut: pointed at the primitive and claimed that \( \text{MAC} = \text{Hash} \) with a key (private key signature), and good encryption will solve it; asked if there is “one pass method.”

• Katz-Y. FSE 2000: first answer YES (under the name “unforgeable encryption”)

• An-Bellare-Rogaway, Bellare-Rogaway, Bellare-Nemprepreprepreprepreprepre, Krawczyk,...
Types of AE

• Krawczyk analyzed Paradigms for separable AE
  – Encrypt-then-MAC: \( C=E(k_1,M), \ H=MAC(k_2,C) \) and send both
  – MAC-then-Encrypt: \( H=MAC(k_2,M), \ C=E(k_1, M \| H) \) and send C
  – Encrypt-and-MAC: \( C=E(k_1,M), \ H=MAC(k_2,M) \)

• All are possible specifically some are generically good (any Mac and any encryption will do)
• Fast solutions one-pass (Rogaway, Jutla, Gligor....etc.)...
• NIST standards....
Encryption via the user: solution

- Use **Authenticated Encryption**: with Encryption field and an Authentication field. Encrypt and MAC (parallel on server side) checking sequential.

- Use **Pseudorandom Function** based encryption:
  - Each display has a large enough “unique context”= seed; No need to extract real randomness (costly);
  - Derive from the seed a random pad;
  - use pad to exor with messages.
  - For more usage (forward looking design): enhance seed with action control in deriving the pad for cryptographic separation/ various length solution
Encryption- in Adx context

• There is a unique tag, and the shared key is a seed of a Pseudorandom function F.
• Since it is unique per auction, every pseudorandom application gives fresh (pseudo)randomness, so we have strong security called: “Semantic Security” (cannot understand the content!)
• \[ C = [F(k1, tag, action) \text{ exor } M], \]
• \[ H = \text{MAC}(k2, M) \]
Advantages

• Fast, does not slow operation!
• Semantic security (due to unique display context tag)
• Flexible utility: F is variable size fields from small to large (for various extensions); authentication only/ encryption only modes can be used.
• Minimal added function (reuse existing/ standardized components whenever possible and research the core new components).
• For security\textarrow The system has crypto engine built in which can be used for other purposes!!!! Can be used to encrypt initial bids if so desired... etc.
Summary for Adx Security

- The system works in this large scale of billions of transactions being encrypted per day (performance tested extensively!!)
- Helped engineering and business!
- The encryptor is essentially: a multi-use, different field size adaptable, enc+auth system.

- (There are other security/crypto/privacy components)
Crypto Designer Goals Achieved

• General utility for privacy and/or integrity for online operation
• Out of band/ SSL/TLS/ etc. in use for key management

→ in ADX/ display ads engineering group: security/ crypto awareness was raised: crypto can solve business issues! Can help engineering!

• Crypto is a friend not a foe!
Indeed.. Extensions came..

• Moving to mobile: need to encrypt certain info of mobile user/ device properties, from Adx to the agencies!
• Encryptor easily extended in no time....

Next: two more contributions:
Privacy and data Liberation

• Adx notifies agencies all info it collects on them

• Adx hides the cookie of users by encrypting them with agency specific keys that Adx keep to itself (the agency does not know the key)
  – This prevents agencies from correlating and finding a common user via the “google cookie”
  – This is a “user privacy issue” solved via encryption
  – If two agencies merge business wise: matching of cookies can be done by Adx!
Verifiability

• Ability to verify correctness of auctions was designed to be very fast (built upon encrypted globally available bidding). Not implemented but disclosed as a possibility to partners.
Summary

• The AdX system has been challenging
• Scale and Speed constraints
  – Security: “Extreme” yet complete Crypto
  – Privacy challenges
  – Only as much as needed
• It posed, both, engineering and research challenges (since 2008):
  – Initial security and privacy solutions have been deployed; while raising the bar for future issues!
  – Future issues became present issues!
There is no fixed recipe for it, just general principles; “results” much less structured than in crypto papers, very few people understand the challenges (rare deployments in general), and getting it right is challenging and satisfying.

Required the right interpretation of the theory

Attack models and risk management apply, incentives for adoption (i.e., business issues related to the recent area “economics of security systems”) and liabilities (i.e., legal issues) apply as well.

Secure components still matter but should be mixed with “added value security” design (the economics behind what the business is investing in).
Differences: theory vs. practice

• Robust Design: Proximity to the system: Requires close interactions w/ engineers, business leaders
• The more “actual solution” is viewed as reducing headaches (enabler), the more credibility and potential future influence.
• Technical clavoyance always helps (is part of the achievement, technical beauty): systems evolve, need to design crypto that is extendible, while current op ongoing.... (true to cloud ecosystems).
• Practice has to be based on solid theory &more..
Thanks!