TWC: Medium: Collaborative: Distribution-Sensitive Cryptography

Principal Investigator: Ari Juels (Cornell Tech)
Principal Investigator: Thomas Ristenpart (University of Wisconsin)
Principal Investigator: Thomas Shrimpton (Portland State University)

Solicitation: `http://www.nsf.gov/pubs/2014/nsf14599/nsf14599.htm`

1 Project Description

Contemporary encryption schemes are almost exclusively *distribution-agnostic*. Their security properties are independent of the statistical characteristics of plaintexts, and they yield ciphertexts that are uniformly distributed bit strings, irrespective of the use case. Distribution-agnostic cryptography is conceptually simple and its generality is often convenient in practice. It fails, however, to meet basic security needs in several important, real-world contexts. To address these needs, and those of applications yet uncovered, we will pursue a line of work centered on what we call *distribution-sensitive cryptography*.

Motivating problems. Our research agenda will target important problems for which conventional cryptography has failed to yield adequate solutions. At first glance, these problems seem unrelated to one another. Our research in DSC, however, will surface deep underlying connections and overlapping practical challenges in problems such as:

- *Brute-force attacks on password-based encryption.* Users tend to select weak passwords. Their password-based encryption (PBE) ciphertexts are thus vulnerable to brute-force password cracking attacks that try to decrypt under guessed passwords and then check if the resulting plaintext is plausible. This problem is serious and pervasive: In the face of today's frequent compromise of mobile devices and cloud systems, PBE is often the last line of defense for highly sensitive data. (Password manager [85] compromise in the cloud is one example arising in practice.)
- *Censorship of encrypted protocols:* Censorship is so pervasive and heavy-handed in some nations that Reporters Without Borders labels them "Internet Black Holes" [79]. Deep-packet inspection (DPI) helps censors identify and block encrypted network protocols such as Tor [1, 20, 24, 57, 58, 86, 87], TLS [13], and Skype VoIP [63, 77]. Anti-censorship tools require encryption primitives capable of producing ciphertexts that appear to be distributed like "benign" cover traffic, at least to a level of fidelity that deceives real DPI-based censorship tools monitoring realistic volumes of traffic. Some existing steganographic tools can achieve provable assurance that ciphertexts match cover traffic distributions [3, 14, 46, 62], but they are impractical for most settings.
- *Securing human-generated authentication secrets.* Users make typos when they key in passwords. Biometrics, such as fingerprints, are noisy. Conventional crypto, however, is fragile in the face of error-prone data. Existing approaches for cryptographic error-correcting codes such as secure sketches and fuzzy extractors [29] seek to address this problem, but leak too much information about low-entropy user secrets to be of practical use.

What is common among these settings is that typical cryptographic approaches fail to account for, or leverage to their benefit, the *distributions* of plaintexts, ciphertexts, and secrets. Decryption with a PBE scheme under the wrong password results in messages that are not distributed like real ones; symmetric encryption does not produce ciphertexts with the distribution of "benign" traffic; and secure sketches and fuzzy extractors cannot capitalize on structure within distributions of the secret data to which they are applied.

Preliminary DSC-style approaches. Recent work by the PIs in two of the three problem domains highlights the promise of DSC-style solutions. PIs Juels and Ristenpart recently introduced honey encryption (HE), a primitive that yields PBE schemes in which decryption with the wrong password outputs plausibly distributed decoy plaintexts. The result is provable security in settings where plaintext distributions can be accurately modeled, even when passwords are low-entropy. While our initial work on HE introduced schemes for some simple plaintext distributions, such as credit-card numbers and prime numbers, extending this work to other plaintext spaces, such as password managers, documents, and so forth, will require design innovations.

In a separate line of work, PIs Ristenpart and Shrimpton introduced a primitive called format-transforming encryption (FTE) [36] for which ciphertexts appear to be uniform samples from a regular [36, 60] or context-free [61] language. Using appropriate languages for network message formats, the resulting ciphertexts can be viewed as steganography that mimics benign traffic only approximately. The resulting schemes, however, are faster than prior steganographic approaches and work against existing real-world DPI systems. Still, FTE only yields individual ciphertexts that mimic benign traffic formatting, and it does not support non-uniformly distributed messages. Thus it is potentially vulnerable to statistical attacks.

While offering promising approaches to addressing the shortcomings of conventional cryptography, our initial work also highlights the scope of the associated challenges and the need for bold conceptual advances in developing and deploying DSC. Instead of pursuing the challenges in isolation, therefore, we have brought together a team of PIs to develop a broad framework for DSC. We will leverage this framework to develop improved security tools in the contexts of PBE and censorship avoidance, and also use it to identify and solve additional problems, such as those arising in the management of noisy secrets.

Unifying framework. We will develop DSC through a principled methodological blend of hands-on empirical study, cryptographic theory, and system design. We view this framework itself as a research contribution capable of complementing and supporting the agendas of other research teams. Our framework will consist of four parts:

(1) *Practice-driven modeling:* A key initial step for any new application will be to experimentally characterize real-world adversarial threats. In the anti-censorship setting, for example, we will study the capabilities of state-of-the-art DPI systems, like those used by censors. At the same time, we will gather or generate datasets to train distribution models and also provide testing data for evaluation. Example data include real-world network traffic for censorship settings, and password leaks like RockYou [76] for PBE.

(2) *Robust, distribution-sensitive definitions:* We will develop formal security definitions that are distribution-sensitive. Generally, this will mean revisiting existing notions and adapting them to the DSC setting. Using the approach of modern provable-security cryptography, we will be able to formally characterize interrelationships among the resulting new definitions, as well as show feasibility and impossibility results. In addition to distribution-sensitive goals, we will also formalize "fallback" security notions that provide best-possible security in case estimates are wrong.

(3) *Practical constructions and implementation:* We will construct distribution-sensitive schemes. These will incorporate models of application-specific distributions, supporting formal proofs of security relative to the new DSC and fallback definitions. Performance will be a key consideration. We will aim to construct practical, easy-to-deploy mechanisms.

(4) *Experimental and formal analysis:* Finally, we will analyze the practicality and security of our constructions. We will build research prototypes of security tools that incorporate DSC techniques, and experimentally evaluate the utility of these prototypes. We will also formally analyze security of our schemes via reduction-based approaches. Typically this will involve some assumption about the gap between the primitive-designer's estimate of a relevant distribution and an adversary's estimate. We will explore new, reductionist approaches to formal bounds on these gaps, as well as empirically validate assumptions via appropriate application-specific experiments, e.g., traffic measurements, analysis of biometric databases, etc.

By developing this framework within the context of several concrete problems, we will not only provide real security improvements in each setting, but also bring to light cross-cutting definitions and tools. One common tool that already emerges is a concept called a *distribution-transforming encoder* (DTE). The lynchpin of the honey encryption construction in [55], a DTE is an encoding scheme whose decoder, given a uniformly random input bit string, yields a distribution close to a target one p_m over a set $\mathcal{M}$. The DSC framework points the way to a natural broadening of DTEs to handle transformations of random variables from one distribution to another, and subsequently to a DTE definition that supports use within honey encryption, FTE, steganography, and other primitives.

This DSC framework also opens up a vista of new cryptographic primitives beyond those identified in previous work. To handle noisy secrets, we propose later a new DSC primitive called a *distribution-sensitive secure sketch* (DSSS). Our exploration of DSC has also led us to recognize that password-based steganography, while used widely in practice, has not received a formal, modern cryptographic treatment. We propose to rectify this gap and, by incorporating use of an appropriate DTE, achieve provable steganographic security even for low-entropy passwords.

Public artifacts and broader impacts. An explicit step in our work will be implementation of DSC tools. These tools will aid our research, but will also serve as a springboard for technology transfer and for impact on security in practice. The PIs have a strong track record of not only releasing public, open-source research systems, but also going the extra mile to help incorporate such implementations into production systems, such as Tor and Google's uProxy. (See Section 6 for more information about our track record in this regard.) We will target similar impact for the proposed work, the ultimate goal being that users of password management systems, activists making use of anti-censorship tools, and others will benefit from the security improvements that our DSC research will provide.

We will by default also make data sets publicly available, the exception being cases in which we have privacy or confidentiality obligations. See the Data Management Plan for more details regarding our handling of data.

Team. Our efforts are cross-cutting, involving data analysis, experimental work, and cryptographic theory. Together, our skills, track record, and momentum uniquely qualify us to pursue the (admittedly lofty) goal of crafting a new approach to cryptographic design. PIs Juels, Ristenpart and Shrimpton have a lengthy track record of successful collaborations in security, cryptography, and system development. Most recently, Juels and Ristenpart have collaborated on honey encryption, and Ristenpart and Shrimpton on format-transforming encryption. We have collective experience

in both theoretical and empirical exploration of modern security artifacts, and a shared vision in which experimental research supports good theory, and vice versa. The PIs also have an extensive history of interdisciplinary research and a mature network of colleagues in other areas that may benefit our work. See the Collaboration Plan for more details about PI backgrounds and our logistical plans.

Proposal Organization and Tasks. In the following sections, we will discuss in turn the three key DSC applications mentioned above, along with three respective solution approaches: honey encryption (HE), a new primitive called Distribution-Sensitive Encryption (DSE) that generalizes and enhances FTE, and Distribution-Sensitive Secure Sketches (DSSS). We will then give further details on our unifying framework, followed by a discussion of the broader impact of our research agenda. Throughout, we identify the concrete tasks we will undertake with the visual call-out **Task**. Our list of tasks serves as a compact outline of what we will deliver (at a minimum); a task schedule may be found in Section 7.

2 Addressing Brute-Force Attacks: Honey Encryption (HE)

Users frequently protect sensitive data using password-based encryption (PBE), applying conventional encryption using a key derived from a user-supplied password P. As noted above, users often choose weak passwords, so a conventional PBE ciphertext C is typically vulnerable to brute-force attack. An adversary may perform trial decryptions of C under guessed passwords until P is identified based on characteristics of the resulting plaintext. As a simple example, if the plaintext is reasonably long ASCII-encoded text, decryption under the wrong key will never yield a valid string with high probability. Given use of authenticated encryption, P may be identified simply because it decrypts C successfully. We refer to this fundamental vulnerability of conventional PBE as the *brute-force barrier*.

PBE is used to protect highly sensitive user data in practice, e.g., credentials in password managers, health-related data, and so forth. Thus the brute-force barrier is a major problem in practice. The common approach of ciphertext hardening via password-based key derivation functions, (e.g., [69]) fails to address the problem. Given typical guessing probabilities of about 1% exhibited by password corpora studied in the wild [10, 11], an adversary can expect to successfully crack one in one hundred ciphertexts on the first try.

Honey encryption (HE) surmounts the brute-force barrier by creating a ciphertext for which decryption under *every* key / password yields a bogus plaintext that still looks valid. Viewed another way, an adversary *cannot tell whether a decryption attempt has succeeded*. As a result, HE gives an *information-theoretic* security guarantee, eliminating the brute-force barrier. If bogus plaintexts are distributed the same as real ones, even a computationally unbounded adversary cannot identify a correct plaintext with certainty.

Building HE with DTEs. In prior work, PIs Juels and Ristenpart introduced DTE schemes for the purposes of honey encryption. Briefly, a DTE scheme consists of a randomized encoding algorithm encode and a possibly randomized decoding algorithm decode. The former maps elements drawn from a set $\mathcal{M}$, called the message space, to strings in $\{0,1\}^\ell$ for some ℓ. We refer to the latter as the seed space, for reasons that will be clear momentarily. Decoding reverses encoding. Suppose p_m is the distribution of messages in some application. A "good" DTE scheme is one such that an adversary cannot distinguish, with more than a small probability, between the pair $(M, \mathsf{encode}(M))$ for $M \leftarrow_{p_m} \mathcal{M}$ (meaning, sample M from $\mathcal{M}$ according to p_m) and the pair $(\mathsf{decode}(S), S)$ for $S \leftarrow\!\!\$ \{0,1\}^\ell$. Intuitively, then, decode behaves as a sampler of messages according to p_m, taking the requisite randomness as input.[1]. Hence the moniker "seed space" for $\{0,1\}^\ell$.

HE can encrypt a plaintext M using, for instance, a simple procedure called *DTE-then-encrypt*. A DTE is applied to obtain an ℓ-bit seed $S = \mathsf{encode}(M)$ and then the seed is encrypted using an ordinary symmetric-key encryption algorithm enc with domain $\{0,1\}^\ell$. The key K used to encrypt the seed may be derived from a password P.

In our initial work on the topic, the security of an HE algorithm $(\mathsf{HEnc}, \mathsf{HDec})$ is defined in terms of a *message recovery* (MR) game in which the adversary attempts to recover M given an HE ciphertext $C = \mathsf{HEnc}_K[M]$, for $K \leftarrow_{p_k} \mathcal{K}$ and $M \leftarrow_{p_m} \mathcal{M}$. MR security applies to settings where an adversary's goal is to recover a full plaintext—as when aiming to compromise credentials for user accounts.

Given a good DTE in the sense described above (and under certain technical conditions), an adversary's probability of winning the MR game is close to the min-entropy of the message distribution. Thus, for example, assuming encryption under a password from a distribution with guessing probability 1%, the success probability of an unbounded adversary is about 1%—in contrast to the 100% success probability of such an adversary against a conventional PBE

[1] Note that the security goal mandates something more. In particular, encoding must also encode a message by picking randomly from all seeds that decode to the message.

ciphertext with a bounded-length password. Additionally, HE adheres to the DSC principle of hedging or robustness: should the DTE be poor, MR security devolves to PBE security for the underlying conventional encryption scheme.

Applications. Given the vulnerabilities created by the brute-force barrier in some of the most sensitive password-encrypted user data, HE holds considerable promise for many applications. We have initiated practical exploration of HE with the following two applications:

Password managers are applications that permit users to encrypt suites of passwords and account information, e.g., website names, under a single password known as a *master password*. The resulting ciphertexts are often backed up in and synchronized through the cloud, where they are vulnerable to bulk compromise and brute-force cracking attacks. The LastPass service has already suffered a breach [85]; a recent study [59] has uncovered a spate of vulnerabilities in the most popular services. PIs Juels and Ristenpart have developed a password manager called SweetPass that leverages HE to prevent brute-force cracking. An ongoing project, SweetPass is slated for release in the near future as an open-source tool.

Genetic information is extremely privacy-sensitive. It can reveal susceptibility to disease and has extraordinary longevity: the disclosure of an individual's genome has not only lifetime impact, but also impact on an individual's relatives and descendants. GenoGuard [48], a system developed by PI Juels and colleagues in collaboration with a geneticist, takes advantage of HE to encrypt sequenced genomes so as to withstand brute-force attacks against password-based encryption.

2.1 Research challenges in HE

While these results hold promise, several major research challenges need to be addressed to lay the groundwork for widespread adoption of these and other HE tools. Our research program will address four major challenges. In Section 4, we discuss DSSS as a potential solution to one of these, the problem of *typo-safety*. The three others are:

Improved DTEs. The security of HE constructions hinges on DTE goodness and yet, as for the examples above, DTEs can be challenging to construct. The SweetPass and GenoGuard systems rely on complex, one-off constructions achieved without the benefit of underlying design principles. An important thrust of this proposal will be the development within the DSC framework of *broadly applicable DTE classes*.

An especially promising such class is based on the principle of *rejection sampling*, a common approach to generating samples according to a target distribution. Consider a randomized algorithm gen that takes as input a bit string from $\mathcal{R} = \{0,1\}^r$ for some r and outputs either a value in a set $\mathcal{M}$ or a distinguished error symbol $\perp \notin \mathcal{M}$. Then to perform rejection sampling using gen, one repeatedly chooses a value R from $\mathcal{R}$ uniformly and runs $\mathsf{gen}(R)$. The routine stops when the output from gen is not $\perp$. We denote this routine as the randomized algorithm RejSam that takes as input $S = \mathcal{R}^\alpha = \mathcal{R} \times \cdots \times \mathcal{R}$ for some number α sufficiently large for gen to yield an output in $\mathcal{M}$ with high probability. A concrete example of cryptographic import is prime number generation, where $\mathcal{M}$ is the set of prime numbers in some range, and R is treated as a bit string representing a random integer in that range. Then gen runs a (randomized) primality test. More generally, rejection sampling can be useful in a number of settings where $\mathcal{M}$ is hard to enumerate directly or sample from according to the desired distribution.

One might consider building a secure DTE for arbitrary rejection sampling procedures using an approach generalized from the DTE for prime numbers given in [55]. Fix an $\mathcal{R} = \{0,1\}^r$ and an associated algorithm gen. Decoding of a string $S \in \mathcal{R}^\alpha$ runs gen on each component of S, and outputs the first output value in $\mathcal{M}$. Encoding of a message $M \in \mathcal{M}$ samples an $S \in \mathcal{R}^\alpha$ uniformly subject only to the constraint that $\mathsf{RejSam}(S) = M$, i.e., samples uniformly from $\mathsf{RejSam}^{-1}(M)$, the preimage set of M. Such encoding can often be accomplished straightforwardly, through explicit inclusion of a value $R \in \mathsf{gen}^{-1}(M)$ in S. Unfortunately, this approach does not work in general.

To ensure low failure probabilities can require large α and, in turn, large encodings. In the case of prime numbers, achieving tight asymptotic security bounds requires α superlinear in r. For $r = 1024$ (the prime length for a 2048-bit RSA key), an example concrete parameterization in [55] is $\alpha = 35,393$, which yields a 4.5 MB encoding for a single RSA private key. Such encoding inefficiency is an impediment to practical use in applications such as HE.

We propose to address this problem by developing a novel DTE scheme for arbitrary rejection sampling settings that achieves only (small) *constant* storage overhead. Our starting point is the use of a pseudorandom number generator (PRG) to stretch a short seed to a larger sequence of pseudorandom bits that can then be used to drive a sampling algorithm. That is, we have $S \in \{0,1\}^k$ for some security parameter number of bits k (e.g., 128) and modify decode to first stretch S to a larger string $\mathcal{R}^\alpha$ using any classic PRG construction. But this approach implies the ability to efficiently compute a PRG seed S, for α polynomial in k, such that $\mathcal{R}^\alpha$ contains $R \in \mathsf{gen}^{-1}(M)$. Such computation can

be shown in general to violate the security definition of a PRG.

We will side-step this challenge by drawing on a technique from simulation-based cryptographic proofs. We will use a special kind of "programmable" PRG construction in a scheme roughly as follows. We use a seed $S = (\sigma, \overline{R}) \in \{0,1\}^k \times \mathcal{R}$. Decoding generates a sequence of inputs for gen as $H(\sigma \| 1) \oplus \overline{R}, H(\sigma \| 2) \oplus \overline{R}, \ldots$ where H is a cryptographic hash function. In words, we use the PRG as before, but now also mask it with $\overline{R}$. The latter will give us the flexibility needed to sample preimages during encoding of a message M, since we can set $\overline{R} = M \oplus H(\sigma \| i^*)$ for appropriately distributed i^* and random σ, needing to check only that no $i < i^*$ is such that $\mathsf{gen}(\overline{R} \oplus H(\sigma \| i)) \neq \perp$. A key challenge will be proving that such an encoding procedure can be made efficient, as well as finding and using suitable assumptions on H to prove security.

Task 1: *We will formalize and prove efficiency and security for a compact DTE suitable for use with arbitrary rejection sampling procedures.*

Beyond MR security for HE. The message recovery security goal for HE proposed and explored in our prior work addresses settings in which attackers must recover the full plaintext. As noted above, this goal makes sense when, e.g., plaintexts are credentials needed by an attacker to compromise a user's accounts. An important question, though, is whether HE can be proven to meet stronger security goals as well. To do so in DSC settings, however, requires new security notions as well as an understanding of fundamental security limits in low-entropy settings.

In ongoing work we have begun exploring new notions. The first is a semantic-security style notion that we call target-distribution semantic security (TDSS). An attacker is given the encryption of an unknown message drawn from $\mathcal{M}$ according to a target distribution p_m. It must predict a predicate (one bit of information) of the encrypted message. We say that a scheme is secure, informally speaking, if it cannot predict this bit with higher probability than when not given any ciphertext at all. As with HE, we seek schemes that take advantage of distribution sensitivity to achieve this goal even when keys have such low-entropy that brute-force attacks can arise. Our initial results show that one can prove the HE scheme from [55] secure in this sense.

A second thrust is understanding nonmalleability for low-entropy key settings. Note that the standard HE scheme described earlier is trivially malleable: an attacker can flip ciphertext bits in a way that yields predictable changes to the plaintext when decrypted. We have formalized a notion of targeted-distribution nonmalleability security that rules out such trivial mauling attacks, even in low-entropy settings. These initial works suggest that there in fact is a whole ecosystem of interrelated security notions when one considers distribution sensitivity and low-entropy keys.

Task 2: *We will map out the ecosystem of distribution-sensitive security notions for HE and devise and analyze constructions under them.*

Empirically-driven adversarial modeling. The difficulty of constructing good DTEs for complex distributions poses a challenge for certain applications of HE as well as other DTE-driven tools that we will explore in this proposal. Natural language offers an example: a good DTE for a natural language document would imply the ability to generate synthetic documents that can fool a human being—a task well beyond the state of the art in artificial intelligence.

Working in our favor, though, is the fact that adversaries' capabilities are bounded by the tools or computational resources available to them. For example, an adversary seeking to automate extraction of sensitive data from compromised documents might reasonably use commercial data-loss prevention (DLP) software, which is designed precisely to identify such data (for redaction or alerting). For adversaries of this limited but realistic class, an HE scheme that encrypts only DLP-identified fields is of potential practical interest. Such an HE scheme would protect the sensitive data likely to be targeted by an adversary while avoiding the complexity of constructing a DTE that models natural language. We further consider such empirically-driven adversarial modeling in the other thrusts of this proposal (e.g., studying DPI tools to develop censorship-circumvention techniques).

Task 3: *We will identify real-world adversarial behaviors and limitations in applications of interest for HE and distill out models that drive provably secure and practical HE constructions.*

3 Circumventing Censorship: Distribution-Sensitive Encryption (DSE)

At a high level, honey encryption is about reproducing a target distribution under decryption. We now turn our attention to the matter of reproducing a target distribution under encryption, taking as motivation the important problem of

circumventing Internet censorship.

Background. For years, nation-states and other network operators have filtered Internet traffic by inspecting IP addresses and TCP port numbers. Citizens subjected to such filtering have employed a variety of tools to obscure TCP/IP information—the Tor network [25], for example. However, the introduction of deep-packet inspection (DPI) technology has enabled more nuanced censorship. By peering into packet payloads, censors may now filter traffic based on what higher-level protocols and application data are being carried. Governments have used DPI to block based on blacklisted keywords appearing in HTTP request headers [21], TLS traffic [13], Tor traffic [1, 20, 24, 57, 58, 86, 87], and Skype VoIP [32, 33, 63, 65, 74, 77].

An obvious response is to apply steganographic tools to packet payloads. However, traditional tools have been ad hoc and offer uncertain security (at best). Those with well reasoned security guarantees [46] provide too little bandwidth for many real-world applications, like surfing the web.

An alternative is to simply obfuscate packet payloads by using conventional encryption on them; after all, ciphertexts should look like random strings. Recent work, however, shows that encrypted (or compressed) payloads are easily detected [84]. In any case, payloads randomized by conventional encryption will not pass whitelist-based filtering, which allow only payloads that conform to particular protocol formats.

Protocol mimicry. Recently, a sheaf of more sophisticated obfuscation techniques have been proposed, largely in support of the Tor Project's pluggable transports architecture [2]. This provides a layer below existing Tor encryption for mechanisms aimed at obfuscating the existence of Tor traffic. Several of these techniques seek to produce traffic that mimics particular applications or protocols. Among these, Skypemorph [64] reformats traffic to look like Skype VOIP traffic. Stegotorus [83] encrypts and (more or less) directly embeds data into fixed templates of HTTP message headers, Javascript objects and PDF files, a la traditional steganographic techniques. Format-transforming encryption [36], which we will discuss shortly, induces a (secretly keyed) mapping from plaintexts to elements of user-specified sets, e.g., strings in the language of a regular expression capturing the structure of HTTP-request messages.

In principle, censorship-circumvention systems that rely heavily on protocol mimicry are fragile in the presence of sufficiently sophisticated censors. Recent work [47] observes that, in the limit, such a system must be able to mimic every aspect of a protocol, even a specific implementation of the protocol, including its quirks and bugs. And the mimicry needs to be sustained against an "active warden," a censor that performs active and adaptive probing of communicating parties.

While sympathetic to warnings raised by this observation, we believe that protocol-mimicry can be of significant use *in practice*, even if the mimicry is far from total. This belief is supported by three observations. First, there is little evidence that real censorship adversaries do, or even can, carry out highly sophisticated mimicry-catching tests at line speed. (Work on implementing the well-known Bro system [68] at the university level might suggest the contrary [34].) Second, it is not known what fraction of real traffic would be flagged as "mimicry" by such tests. Finally, recent work [36] shows that real DPI devices *are* vulnerable to so-called protocol-misclassification attacks.

Format-transforming encryption. In support of the last observation, PIs Ristenpart and Shrimpton recently introduced format-transforming encryption (FTE) [36], which we can now view as a kind of DSC primitive that is well suited for the payload obfuscation task. This is because FTE enables precise control of the format of ciphertexts, i.e. how they will appear to an observer on the wire.

The FTE system built in [36] used the fact that modern DPI systems seem to rely heavily on regular expression matching, or ad hoc tests that are reasonably approximated by regular-expression matching, in carrying out traffic classification. In particular, our FTE system allowed users to specify the desired format of the (ciphertext) packet payloads by inputting a regular expression. Subsequently, the FTE ciphertexts were guaranteed to be random strings from the language described by that regular expression.[2]

In [36], we showed that a battery of six modern DPI systems (including an commercial, enterprise-grade hardware DPI system) can be caused to classify FTE ciphertext messages as any protocol that we chose, no matter what was the actual underlying plaintext. This was done by providing FTE with appropriate regular expressions. In the simplest cases, these were lifted directly from the DPI system, or created manually using knowledge of protocol RFCs. In other cases, the regular expressions were learned, by a simple procedure, from network traces.

Moreover, we were able to set up an FTE-powered client inside of China, and tunnel arbitrary plaintext traffic

[2]Follow up work [61] gave new algorithmic techniques that admitted more efficient regular-expression-based FTE, as well as expanding the scope of formats to cover context-free languages.

(including Tor, and HTTP traffic from banned sites) to and from an FTE-powered proxy in the USA.[3]

We note that FTE is one of the few proposed obfuscation techniques actually integrated into the Tor distribution as a pluggable transport, and the only one that targets both blacklisting *and* whitelisting filtering rules.[4]

3.1 From FTE to DSE

The initial successes of FTE are encouraging because they show that regular-expression-based DPI can be fooled, and by systems that also support typical Internet usage (ie., watching YouTube videos). Moreover, our investigations suggest that regular-expression-based DPI is the norm in practice, as do some manufacturer documents, e.g. [9].

That said, FTE targets only the goal of payload obfuscation, and achieves this producing ciphertexts that mimic *individual* protocol messages. It makes no effort to respect properties that would be expected over an ensemble of real protocol messages, i.e. distributional issues. Fundamentally, this is because FTE ciphertexts are *uniform and independent* samples from some language of strings. Thus, a *sequence* of FTE ciphertexts is unlikely to display, say, a distribution of message lengths that mimics reality. This shortcoming may foster detection by traffic analysis attacks. Moreover, and in contrast to conventional encryption, formatted ciphertexts have observable structure by design. As a result, DPI may come to support more sophisticated statistical attacks based on protocol message structure or content.

Modeling real adversaries. We do not yet know if these kinds of attacks are a real threat, now or in the near future, because the community lacks understanding of real-world DPI tools and their capabilities at scales of interest.

We do know that operating at wire speed places significant constraints on what computational tasks can be carried out. Consequently, nation-state censors seem to perform two-stage processing. The first stage operates at wire speed, where DPI is used to select out "suspect" traffic for further, more computationally intensive analysis in the second stage. As we've already said, there is some evidence to suggest that regular-expression matching forms the core of wire-speed processing at nation-state traffic volumes. But we do not in fact know how accurate this is for current DPI, nor how predictive of the future. (For example, if regular-expression-based tests are the current norm, is it a small step, or a large jump, to support tests requiring full payload parsing?) This brings us to our next task:

Task 4: *We will explore the performance, capabilities and potential configurations of DPI products (e.g., Blue Coat ProxySG and Packetshaper) known to be used in censorship.*

Documentation for commercial DPI tools is typically not directly disclosed by manufacturers. PIs Ristenpart and Shrimpton, however, along with other members of the circumvention research community and a major industrial partner are also part of a burgeoning effort to establish a DPI lab allowing direct experimental access to a number of systems. To see where DPI systems may be going, we will also look to research efforts in this space, and seek collaborations where appropriate with colleagues at our respective institutions with relevant expertise.

We will also carry out a measurement study, using traffic from our three respective institutions as well as external sources, to empirically characterize what "benign" traffic looks like, along various dimensions. (PI Ristenpart already has a full-packet-capture network tap in place at the University of Wisconsin. All use of it is guided by appropriate IRB exemptions or reviews.) This corpus will also support investigations of efficacy of distinguishing attacks on FTE, DSE and other obfuscation methods. All this will require some engineering effort to develop software analysis tools for simulating attacks. The resulting software framework will be made available for other researchers to use.

These empirical investigations will help to shape formal models of DPI-powered adversaries. In turn, these formal models will enable principled security arguments and, in some cases, formal proofs about future designs. Our starting point is prior work by Hopper, Langford and von Ahn [46]. They provided the first formal study of steganographic systems, including giving semantic-security style notions of security. Their security notion is very stringent, assuming that the adversary (in this case, the DPI-powered censor) has complete knowledge of the distribution of traffic on the channel at all points in time. While conservative, this seems unfounded in practice, especially at nation-state volumes of traffic. What's more, because the adversary is so powerful in their notion, they are able to prove security only for inefficient schemes. Establishing models and security notions that more accurately reflect the realities of real DPI-powered censors should have the side-effect of allowing us to prove security for more efficient schemes.

Task 5: *Drawing on complexity-theoretic characterizations of language recognition (e.g., circuit complexity), we will seek to model formally and concretely the space of traffic classifiers (steganography detectors) implementable by a space- and computation-bounded censor. We will aim to construct DSE schemes that shape traffic that is provably undetectable within the resulting models.*

[3] The client ran tests every five minutes for one month, with no sign of detection by the GFC. After one month, we shut it down.

[4] The other fielded pluggable transports rely on payload randomization.

Generalized DTEs and DSE. We will also set ourselves to building the next-generation of steganography primitives that are fast, yet secure relative to real-world adversaries. Towards this end, we will start by generalizing a primitive from the study of honey encryption, namely DTEs. Consider two sets $\mathcal{X}, \mathcal{Y}$ equipped with distributions p_x, p_y respectively. A *generalized DTE* would be tasked to encode p_x-distributed samples of $\mathcal{X}$ so that they appear to be p_y-distributed samples of $\mathcal{Y}$, and vice versa. Setting $\mathcal{Y}$ to bitstrings and p_y to uniform, and $\mathcal{X}$ to prime numbers and p_x to uniform gives the DTE we proposed previously [55].

This generality now allows us to recognize the primary FTE constructions as an instance of what we call *encrypt-then-DTE*. Setting $\mathcal{X}$ to bitstrings and p_x to uniform, and $\mathcal{Y}$ to some target language with distribution p_y, one could use encrypt-then-(generalized-)DTE to build FTE schemes that turn plaintext into ciphertexts that are samples from $\mathcal{Y}$ that appear to be p_y-distributed. Indeed current FTE is encrypt-then-DTE with a DTE whose outputs are uniform samples from a specified regular or context-free language. Being able to treat non-uniform distributions, would be a significant step forward: it would specifically help to defuse statistical attacks by DPI adversaries that look for uniform distributions. We call the resulting approach distribution-sensitive encryption (DSE), which we view as an efficient kind of steganography that need not fool human observers.

The encrypt-then-DTE viewpoint on DSE proves to actually capture prior work on steganography as well: Cachin proposed what amounts to an encrypt-then-DTE construction, where the DTE was based on universal coding schemes, to achieve information-theoretically secure steganography [14]. A similar approach was taken in follow-up work by Backes and Cachin for public-key steganography [3]. As far as we are aware, this work has not yet led to any practical mechanism.

To achieve generalized DTEs, one might hope for schemes that work for arbitrary sets $\mathcal{X}, \mathcal{Y}$ with respective distributions p_x, p_y. Unfortunately, it is not hard to see that not all pairs of sets/distributions can be supported. Correctness mandates the ability to invert, so building a DTE that maps from a high-entropy distribution to a very low-entropy one will, in general, be information-theoretically infeasible. Generalized DTEs will provide a principled way to explore bandwidth/security trade-offs and, ultimately, lead us to understand the foundations of these trade-offs.

Task 6: *We will formalize generalized distribution-transforming encoders (DTEs) and appropriate security notions. We will investigate fundamental trade-offs between efficiency and security, providing formal bounds showing the impossibility of certain combinations of input/output distribution pairs.*

With these boundaries in place to help guide us, we will turn to building generalized DTEs, particularly ones suitable for the anti-censorship setting. Guiding this search will be the the insight that in many cases an imperfect approximation of the covertext distribution will be sufficient, as per our adversarial modeling discussed above.

Task 7: *We will explore constructions of generalized DTEs for practical steganographic encryption. We will explore what can be done efficiently in the censorship circumvention setting, in particular relative to realistic attack models (Tasks **4**,**5**).*

An oft-suggested mechanism for protocol steganography is tunneling (c.f., [47]). That is, given a full implementation of some cover protocol, one can submit encrypted data as protocol messages to the implementation. This would seem to provide very high-fidelity mimicry, at least for one implementation and when the tunneling protocol is lossless. Even so, tunneling may still admit detectable discrepancies with regards to normal use of the implementation. For example, if the implementation applies lossless compression to incoming messages, then tunneling in this fashion will lead to larger-than-normal protocol transcripts.

Task 8: *We will explore the use of tunneling as a DTE, comparing its efficiency and security to our de novo DTE constructions.*

Mimicry above the message-level. Finally, we note that we have been discussing statistical attacks on FTE at the level of protocol messages. Censors with resources sufficient to maintain a greater amount of state could mount statistical attacks at the connection level, e.g., checking that a sequence of upstream and downstream FTE ciphertext protocol messages observe expected semantic patterns. We do not yet understand how much state would be required to mount connection-level analysis at nation-state traffic volumes, although we will explore this in our evaluation of real-world DPI adversaries (see Task **4**). We will use what we learn to push distribution matching behaviors "up the stack" in a way that retains efficiency.

Task 9: *We will explore stateful DSE, where the distributions to be matched are a function of state including previously exchanged messages—both the mimicked messages and the actual plaintexts.*

A natural way to approach this is to build sender and receiver state machines that determine which distribution will be used for the next message. Task-structured probabilistic I/O automata [15] (or more general probabilistic automata [78]) may provide a convenient formal model for describing these stateful behaviors.

Stateful DSE can also take advantage of published state-machine models of various protocols. They also allow for things such as mimicking the number (and type) of connections over which to send protocol messages.

Finally, we note that working above the message level also provides opportunities for dealing with active adversaries, which may send probing messages to possible proxy servers in order to see how they respond. The Great Firewall of China is known to employ such tactics to help find and blacklist Tor servers [87]. The automata can include transitions to error states that result in innocuous behavior when malicious messages are detected (e.g., by a message authentication failure when the client and server share a key).

3.2 Low-entropy Steganographic Security and Password-based DSE

The previous discussion of distribution-sensitive encryption was strongly focused on the real-world censorship circumvention problem. Let us step back now to consider a broader perspective on steganography and new opportunities for DSC to have positive impact.

A classic approach for steganography attempts to hide a conventionally encrypted message in the low bits of pixel data in an image (c.f., [16, 19, 50]). We first observe that this can be viewed as an encrypt-then-DTE construction, just like the DSE schemes discussed in the last section: here the DTE scheme is parameterized by one or more images, and encodes by substituting ciphertext bits into appropriate locations in order to create the covertext. The security target for image-based steganography is most often ad hoc, essentially that human observers cannot distinguish between the original image and the one with ciphertext embedded. But this target leads to weak schemes, as simple statistical tests or other algorithmic approaches can easily detect the presence of an embedded ciphertext [16, 70].

Compounding this is the fact that, as with conventional encryption, many practical uses of conventional steganographic tools rely upon human-memorizable passwords. This makes them vulnerable to offline, brute-force attacks that not only confirm the existence of hidden content but also *reveal the password and message*, as suggested by Provos and Honeyman [70]. The procedure is as follows. Given a possible covertext C, run decryption with each of the plausible passwords. Should none decrypt to properly formatted plaintexts then one can conclude correctly, with high probability, that C is in fact a benign message. Otherwise, C is indeed a covertext and the message and password are revealed. The format might be correct ASCII encoding of text or the like. In a chosen-message attack setting, including the steganographic security definition formalized by Hopper et al., the attacker knows not just the format, but the entire plaintext.

A priori, it may seem that secure steganography with low-entropy keys is impossible. But in specific settings, message distributions are known (or can be well estimated), a fact we will seek to leverage. As a concrete example, and one that is also potentially important in the censorship setting, we will consider password-protected, steganographic key distribution. Here, the goal is to transmit the public key of an asymmetric encryption or signature scheme to another party, while hiding the exchange. In this case, the messages come from a clearly-defined distribution, such as an element sampled uniformly from a group suitable for discrete-log-based schemes, or a uniformly sampled RSA modulus and public exponent.[5] Crucially, the coins used to sample from these distributions should be unknown to parties external to the exchange protocol.

Task 10: *We will formalize security notions for steganographic systems when used with low-entropy keys such as human passwords.*

Specifically, we will develop security notions of *low-entropy steganographic security*, wherein covertexts must be indistinguishable from non-steganographic "benign" channel messages, even when low-entropy keys are used. The key to achieving such a notion, as we've just suggested, will be to target security when the challenge messages are drawn from a distribution with coins unknown to the attacker. We will explore settings in which the attacker and scheme designer know the distribution, as well as (arguably more practical) settings in which they are only able to each obtain some number of samples drawn from it.

New steganographic constructions. We will also pursue the design of constructions that meet this new goal of

[5] Interestingly, we expect that the latter will be very challenging, in fact it may only be feasible with machinery such as cryptographic obfuscation [43]. The reason is that it would seem to require having a sampling routine that produces a RSA modulus without knowing the factors.

low-entropy steganographic security. This will not be a simple matter of tweaking existing schemes that are provably secure when given high-entropy keys. For example, our preliminary work shows that the schemes from Hopper et al. [46] are vulnerable, in the low-entropy key setting, to a brute-force attack similar to the one described above, even when the underlying message is a uniform bit string. (The weakness in these schemes is the use of redundancy to ensure low probability of decryption errors.)

Task 11: *We will show an attack against the low-entropy steganographic security of the Hopper et al. scheme for uniformly selected random messages.*

Our initial approach will be to extend the encrypt-then-DTE construction to include the techniques from HE. That is, we will first apply a DTE tailored to the plaintext distribution, then apply a carefully selected password-based encryption scheme, and finally, to obtain the ciphertext, apply to the intermediate ciphertext a DTE tailored to the cover distribution. We call this a *DTE-encrypt-DTE* construction. When the plaintext and ciphertext distributions are amenable to building DTEs, such a construction has the potential to provide steganographic security (the attacker can't determine if it is an encryption or just a "benign" message), *even* in the face of brute-force attacks like the one described above. The reason will be that for any password, decryption of a message, whether it be benign or the output of DTE-encrypt-DTE, will produce a plausible plaintext message. DTE-encrypt-DTE is just one framework for building low-entropy steganography (and DSE), and we expect that our work will uncover others.

Task 12: *We will develop new approaches to low-entropy, and especially password-based, steganography.*

4 Securing Human Secrets: Distribution-Sensitive Secure Sketches (DSSS)

HE, DSE and password-based DSE treat the problems of reproducing target distributions for messages, ciphertexts, and both messages and ciphertexts, respectively. We now consider the orthogonal problem of managing a secret K sampled from a target distribution. Of particular interest are noisy, human-generated secrets such as biometrics and (possibly mistyped) passwords. We consider the specific problem of deriving an error-detecting or error-correcting value that helps deal with such noisy secrets, yet leaks as little information as possible. Applying the DSC framework to this challenge yields a new concept that we call a *distribution-sensitive secure sketch* (DSSS).

Problem setting. Users often make typographical errors or misremember "something-you-know" secrets, such as passwords and answers to life questions. Similarly, biometrics, or "something-you-are" authentication values, are inherently noisy; e.g., fingerprint images vary due to variability in pressure, rotation, skin condition, and so forth.

Systems that manage user credentials securely in explicit form can correct errors by explicit comparison. A server that stores a fingerprint template K^* can compare it against a fingerprint reading K proffered by a user at login, accepting K as valid if K^* and K are close under some suitable metric. Conventional cryptographic primitives, though, are specifically designed to be brittle in the face of noise and actually prevent direct comparison. Flipping a single bit of input, for example, yields a completely different output value when using a well-designed hash function.

Such intolerance to noise or error often impedes secure and usable system design. For instance, biometric templates *cannot be protected via hashing*. Similarly, hashed passwords offer no *typo-safety*. In hashed form, a valid password is indistinguishable from an incorrect one. In honey encryption and password-based DSE, lack of typo-safety is especially problematic: typos induce incorrect (but plausible-looking) decryptions that can mislead legitimate users.

Existing approaches. *Fuzzy cryptography* [12,27–29,42,52,53] is one approach to reconciling conventional cryptography with noisy secrets. A foundational tool is a *secure sketch* (the underpinning of a tool known as *fuzzy extraction*).

A secure sketch is a pair of efficient probabilistic algorithms $(\mathsf{SS}, \mathsf{Rec})$ that operate over a secret space $\mathcal{K}$. The procedure SS takes an input secret $K \leftarrow_{p_k} \mathcal{K}$, where p_k is a probability distribution over $\mathcal{K}$; it outputs a "helper" string $s \in \{0,1\}^*$. The procedure Rec inputs a (possibly corrupted) secret $K' \in \mathcal{K}$ and helper string s and yields an output in $\mathcal{K} \cup \perp$ (where $\perp$ indicates a decoding failure). An $(e, \tilde{e}, t)$-secure sketch is one for which: (1) Recovery is possible for small errors, i.e., $\mathsf{Rec}(K', \mathsf{SS}(K)) = K$ if $\mathsf{dist}(K, K') \leq t$ for some distance metric dist and (2) The information leakage of the helper string in terms of average min-entropy is small. Specifically, if $\tilde{H}_\infty(p_k) \geq e$ then $\tilde{H}_\infty(p_k \,|\, \mathsf{SS}(p_k)) \geq \tilde{e}$.

A secure sketch may be viewed as a systematic error-correcting code on K. What distinguishes it from an ordinary such code is the bound (property (2)) on information leakage produced by the redundancy s of the code.

The security definition for a secure sketch is parameterized by the average min-entropy of the secret distribution p_k. But existing *secure sketch constructions lack distributional sensitivity*. For example, Dodis et al. [29] describe a secure sketch for edit distance, a measure apt for, e.g., typos in passwords. Their construction is agnostic to the

underlying distribution, and thus achieves only loose (and only asymptotically bounded) security $e' = e - t \cdot (\log F) \cdot \exp(O(\sqrt{\log(n \log F) \log\log(n \log F)}))$, where F is the alphabet size for characters of strings in $\mathcal{K}$, and n their length.

As a simple example, a construction with $t = 2$ on eight-digit passwords over ASCII characters would have an average min-entropy bound of at least $t \log F = 16$ bits (excluding the exponential asymptotic factor and its potentially large constants), but the average min-entropy of a typical password distribution in the wild [10, 11] is just under 7 bits!

Such drawbacks, as well as a general inability by the research community hitherto to match theory to real applications, has meant that secure sketches (and fuzzy extraction) have seen little or no use in practice.

Distribution-sensitive secure sketch (DSSS). The DSC framework points the way to a dramatic reduction in entropy loss that can bring fuzzy cryptography into the realm of practicality. The result is DSSS, a secure sketch that is engineered for a specific target secret distribution.

As a specific focus and example, we now consider a DSSS construction for typographical errors in password entry that promises to improve greatly on the edit-distance secure sketch of Dodis et al. for this application. The goal is a secure sketch $(\mathsf{SS}, \mathsf{Rec})$ for a distribution p_k of user-selected passwords, i.e., over passwords $K \leftarrow_{p_k} \mathcal{K}$, where $\mathcal{K}$ is the set of permissible passwords in a system. The secure sketch should permit password recovery given a user-typed password K'—possibly with some typos—when $\mathsf{dist}(K, K') \leq t$. Here, dist is edit-distance, a standard metric for typographical errors, and t is a system parameter that would in practice be fairly small, e.g., $t = 2$ or 3.

In the case of passwords, it is possible to leverage two key observations to achieve a practical DSSS. First, relatively large outputs are often acceptable in a password DSSS. For example, adding 10 KB—or even 100 KB—of secure sketch data to protect a ten-character master password in a password manager on a laptop is eminently practical, even though it creates a more than 1,000x or 10,000x expansion relative to the password itself. Thus it is possible to base a secure sketch in such settings on an extremely low-rate error-correcting code, potentially an unorthodox, application-specific one. Second, popular user-selected passwords studied in the wild have a high average pairwise edit distance. In order of popularity, here are the top twenty passwords leaked from the RockYou data set that meet the (broadly representative) password composition policy of iCloud:

`Password1`	`Princess1`	`P@ssw0rd`	`Passw0rd`	`Michael1`
`Blink182`	`!QAZ2wsx`	`Charlie1`	`Anthony1`	`1qaz!QAZ`
`Brandon1`	`Jordan23`	`1qaz@WSX`	`Jessica1`	`Jasmine1`
`Michelle1`	`Diamond1`	`Babygirl1`	`Iloveyou2`	`Matthew1`

One pair has edit distance 1 (`P@ssw0rd`, `Passw0rd`) and another pair has edit distance 2 (`Password1`, `Passw0rd`), but the vast majority have edit distance at least 3. Thus the set of frequently used passwords itself largely constitutes a *good codebook for edit-distance*. The construction in [29] provides no way to make use of this essential insight.

To illustrate how these two observations can give rise to a practical DSS for the special case of passwords, suppose that the complete set $\mathcal{K}$ of passwords selected by a given population of users had minimum edit distance 4. Then a simple DSSS for $t = 2$ would be the pair $(\tilde{\mathsf{SS}}, \tilde{\mathsf{Rec}})$, where $\tilde{\mathsf{SS}}(K) \to \mathcal{K}$ and $\tilde{\mathsf{Rec}}(K')$ maps K' to the closest $K \in \mathcal{K}$ by edit-distance. In practice, given the major password leaks such as the RockYou breach, $\mathcal{K}$ may be effectively viewed as public knowledge. Thus this DSSS has *no conditional entropy loss*, i.e., $\tilde{H}_\infty(p_k) - \tilde{H}_\infty(p_k \,|\, \tilde{\mathsf{SS}}(p_k)) = 0$. In practice real password spaces will not have all passwords as far apart as four, a challenge we discuss below.

Our two observations regarding passwords apply in other key settings (e.g., biometrics) and form an important foundation for exploration of DSSSs. They illustrate respectively the two following general design principles:

DSSS Design Principle 1: Use of low-rate codes: Unlike typical settings for error-correcting codes, in many settings of interest for DSSSs, it is feasible to use a very low-rate code, i.e., a code with high redundancy.

DSSS Design Principle 2: Use of embedded codebook structures: Target distributions of interest (e.g., biometrics and passwords) for DSSSs typically have high average pairwise distance among high-probability secrets. Sets of such secrets may be leveraged as codebooks to facilitate error-correction / error-detection. We refer to such sets of secrets as *embedded codebook structures*. This design principle gives rise to the following definitional retooling task:

Task 13: *We will develop formal definitions and a general framework for identifying and quantifying the utility of embedded codebook structures in target message distributions for DSSS constructions.*

Task **13** is challenging because of the messiness of real-world message distributions. Recall in our example above that many password pairs will have low edit distance (e.g., (`P@ssw0rd`, `Passw0rd`)), so not all of $\mathcal{K}$ can serve as the codebook. A challenging optimization problem results: $\tilde{\mathsf{SS}}$ must output a subset of $\mathcal{K}$ that is pruned to eliminate low-distance pairs but still encompasses as many high-weight, i.e., popular, passwords as possible. To take a graph-theoretic perspective, consider a graph $G = (V, E)$ in which nodes correspond to passwords in $\mathcal{K}$ and an edge (K_i, K_j)

is present if $\mathsf{dist}(K_i, K_j) < t$. Intuitively, to yield a suitable codebook, SS must output an independent set $\tilde{V} \subseteq V$; to preserve conditional average min-entropy, $\tilde{V}$ must have close to maximal cumulative weight under p_k. The maximal independent set problem is NP-hard, as are weighted variants, so practical solutions must rely on heuristics.

Another challenge is that the set $\mathcal{K}$ of possible user-selected passwords is of vast size and not explicitly storable. A second challenge in constructing a DSSS for this example application is handling relatively rare passwords that might not appear in an explicitly stored password list. In general, we will undertake the following task:

Task 14: *Building on DSSS Design Principles 1 and 2, we will use the DSC framework to explore the construction of DSSSs for a wide range of settings of interest, including passwords and biometrics. We will leverage knowledge obtained in specific applications to surface foundational concepts, such as embedded code structures.*

Our DSC framework will guide empirical validation in this task. As the problems underpinning our DSSS design approach are NP-hard, our constructions will rely on approximation heuristics developed by the research community for related problems, e.g., [6, 44]. Our analysis of our DSSS constructions will thus require validation of these heuristics against real-world password data. Specifically, we will experimentally measure the conditional entropy yielded in our DSSS construction according to the formal security definitions for secure sketches.

DSSS for typo-safety in HE. Because HE has the property that decryption under an incorrect key yields a valid-looking plaintext, a problem arises in the use of HE with noisy or error-prone keys / secrets (e.g., biometrics or user-typed passwords). An error in a decryption key can present a legitimate user with an incorrect plaintext that the user may accept as valid. Despite promising proposed solutions, e.g., use of security skins [23] and online testing for the case of password managers, this problem represents a major issue in the practical use of HE. DSSSs offer a potential solution.

In the setting of password-based HE, this issue of noisy secrets emerges specifically as a problem of *typo safety*, namely the ability to detect a mistyped password and trigger a decryption failure. It would be very helpful in practice to have a mechanism that provides typo safety while *causing only minimal conditional entropy degradation for an HE password*.

A DSSS for edit-distance over passwords provides exactly this property. Additionally, in practical settings of interest, such as the SweetPass tool described above, it suffices to detect rather than correct errors. Given a code with minimum distance d in a DSSS construction (e.g., $\tilde{V}$ in the graph-based approach), error-correction can correct up to $t = \lfloor d/2 \rfloor$ errors in general. Error-detection, though, permits detection of a much improved $t = d - 1$ errors using the same code.

Honeywords as DSSS. DSSS also provides a new perspective on the honeywords system of Juels and Rivest [51]. That system stores for each user a list of (hashed) passwords $s = \{K_1, \ldots, K_n\}$, of which K_j is the true password for a random $j \leftarrow_R [1, n]$. The other passwords are fakes selected from a model distribution p_d, i.e., $K_i \leftarrow_{p_d} \mathcal{K}$ for $i \neq j$. The index j is stored in an isolated system called a honeychecker. When a user proffers a password K at authentication, if $K = K_i$ for $K_i \in s$, the index i is sent to the honeychecker for verification. An adversary that breaches the system and learns s still faces the challenge of guessing the right index j; a wrong guess signals the adversary's compromise of s.

The list s may be viewed as the helper string in a DSSS $(\widetilde{\mathsf{SS}}, \widetilde{\mathsf{Rec}})$ with $t = 0$. An ideal password model, i.e., $p_d = p_m$, yields a $(e, \tilde{e}, 0)$-secure DSSS with $e - \tilde{e} = \log n$. (Indices may be replaced with values yielded by fuzzy extraction, to complete the conceptual connection.) If p_d poorly models p_m, less security is obtained. This new DSSS-based perspective promises to yield innovations in honeywords construction. For instance, combining our "programmable" PRG with a model for sampling p_d yields a novel representation s of *constant size for any n*.

5 A Unifying Framework for Distribution-Sensitive Cryptography

In the previous sections, we have described a number of applications for which we will develop DSC techniques. From each thrust, cross-cutting methodologies emerge, including practice-driven threat modeling, empirical estimation of relevant distributions, and provable-security design. Our efforts will take advantage of these themes and developing a unified methodological framework for DSC. It is likely to evolve as we gain further experience while exploring applications; here we provide further details on what we expect to be its key ingredients.

(1) Practice-driven modeling: A key enabler for DSC is the ability to estimate distributions of interest, such as message spaces, covertexts, passwords, and so on. This often requires obtaining access to, or gathering from scratch, appropriate data sets (password leaks, captured network traffic, etc.), methods for compactly representing these empirically observed distributions, and statistical approaches to analyzing them. A second part of this step is distilling out

realistic threat models to ensure that our work addresses attacks of real-world import. For example, we will investigate real DPI system capabilities and assess experimentally the viability of various attacks, as discussed in Section 3.1.

(2) Robust, distribution-sensitive definitions: DSC primitives require new formal security notions. For example, we have new target-distribution message recovery, semantic security, and nonmalleability notions for HE, each an adaptation from previous distribution-agnostic notions. For steganography, we have a head start with the pre-existing security notions from Hopper et al. [46], but adaptations will be needed in the low-entropy key case. And we are able to adapt the target-distribution message-recovery notion to apply to secure sketches. A common recipe emerges: revisit previous notions of security and replace adversarially chosen messages or message distributions with draws from a specific distribution known to the protocol designer. This leads to a new viewpoint on security definition development.

A potential problem with DSC is when the scheme ends up used with messages that are not distributed as expected by the designer, or when adversaries have unexpected side-information about messages or secrets. In such cases, our DSC security goals may not be achievable. We therefore also explicitly target robustness: in addition to these distribution-sensitive goals, we will also formalize appropriate, best-possible "fallback" security notions. For example, for HE the fallback notion is a variant of semantic security suitable for PBE [4] and for steganography we seek standard confidentiality and authenticity as in authenticated encryption.

(3) Practical constructions and implementation: Given empirical models for a target application and corresponding formalism, we then seek constructions for an appropriate DSC primitive. A key challenge is producing compact and computationally efficient message-distribution models for incorporation into these primitives. We will draw on a variety of established modeling techniques, including regular expressions, Markov models, probabilistic context-free grammars, special-purpose sampling algorithms (e.g., prime number samplers)—most of which we have begun exploring in prior [5, 36, 55, 60, 61] and ongoing work [17, 48]—and also appeal to other new application-driven techniques.

(4) Experimental and formal analyses: We will target constructions that are both practical and have demonstrable security properties. For the latter we will blend empirical work with provable security analysis, relative to our newly formulated DSC notions and their associated fallback notions.

Let us provide a few more details. Given sound underlying cryptographic primitives, the theorem statements resulting from these analyses will ensure security when the measured quality of distribution estimates falls within certain ranges. (When they do not fall in this range, security will revert to the appropriate fallback notions.) This proof approach enables us to restrict attention specifically to the difference in quality of the DSC-primitive distribution estimates, and those of an adversary. We will give confidence in these estimates using two approaches.

When possible, we will employ measures of the *absolute quality* of DSC-primitive distribution estimates relative to real distributions. This will work when the distribution has some well-understood and relatively precise characterization. An example is uniform prime numbers in our prior work on HE [55]. Additionally, we will investigate the goodness of a primitive in terms of the *relative quality* or *gap* between the attacker's and defender's knowledge of the true distribution or between the quality of the modeling techniques they employ. This will be useful when distributions can only be captured empirically, as with human-generated distributions (passwords, biometrics, English text), or complex covertext protocols, such as those seen by network operators on the Internet. While empirically-driven modeling is challenging in many DSC settings, an adversary confronts exactly the same challenges in distribution estimation as the designer of a DSC primitive. Thus, even if a defender designs a primitive that does not exactly match real-world data, an attacker may have little usable advantage in attacking the scheme.

A key challenge is quantifying that advantage. We can do so empirically for various specific adversarial strategies. To obtain broader guarantees, we will also explore *reductionist* security bounds similar to those provided traditionally in provable-security cryptography, but here based on hard problems from distribution estimation or learning theory. The target will be theorem statements of the following form. A computationally efficient adversary's ability to distinguish well between a DSC scheme's estimates and a real distribution implies either: (1) a modeling tool for the real distribution that is superior to (e.g., faster or more space-efficient) the state of the art, or (2) an adversarial model that is informed by superior access to real data. For example, an adversary with a significant distinguishing advantage against a DTE based on state-of-the-art password cracking (e.g., [17]) can be translated into a significant advance in password cracking technology. So assuming adversaries and designers have similar access to the real-world distribution, such reductionist security results will rule out successful attacks unless an adversary achieves an unexpected algorithmic breakthrough.

In cases where crisp security reductions are elusive, our estimates of adversarial capability will be best-effort and

experimental. That is, we will provide experimental quantification of how well various adversarial strategies (such as machine learning approaches, statistical tests, etc.) can be made to work.

Task 15: *We will develop a unified framework for analysis of DSC schemes that blends techniques from provable-security cryptographic theory, distribution estimation, and empiricism.*

An inclusive open-source library. As a final contribution, we propose to bring together the various DSC scheme implementations within a single open-source library, tentatively named `libdsc`, that provides easy-to-use APIs for our new cryptographic primitives, as well as lower-level access to the underlying mechanisms for researchers. Here we will build off the PIs' previous experience building similar libraries such as `libfte` [60].

Task 16: *We will implement an open-source library around our various DSC constructions to support experimental evaluation and impact.*

6 Curriculum Development Activities, Outreach, and Result Dissemination

Curriculum development. A key aspect of our work will be in improving curriculum to ensure students have the right skill sets to tackle real-world problems using techniques from applied cryptography.

PI Juels is developing a new masters-level security course at the new Cornell Tech campus over the course of Fall 2014; given the industry-focused mission of the campus, this course aims to impart security principles and knowledge, including applied crypto, specifically to budding technology entrepreneurs. He is also mentoring students in the campus's Company Projects program, guiding them in security problems brought to the campus by industry partners.

PI Ristenpart will integrate DSC-style viewpoints into his teaching at University of Wisconsin. For both a PhD level and a senior undergraduate level, he will prepare appropriate lectures covering topics pertaining to the research conducted in this grant. The particular educational goal will be to encourage students to learn how to think, in a principled fashion, about contexts when performing security mechanism design.

PI Shrimpton will develop a course in cryptography suitable for undergraduates. This course will focus less on proofs and formalisms than his graduate course, and more on applications and developing a good sense of cryptographic hygiene. Development of this course is partially motivated by local industry needs. In fact, Shrimpton will work with Dr. Jesse Walker of Intel to develop this new course.

Outreach and diversity. We believe that scientific communities are most productive when they include researchers from a wide variety of backgrounds: science disproportionately benefits from a diversity of viewpoints. Towards this end, we will make an effort when attracting students to especially target women and underrepresented minorities. The PIs have already had success in this regard. Shrimpton has mentored Mrs. Tashell Kelly (undergraduate), Ms. Morgan Miller (MS 2010), Ms. Erin Chapman (MS 2012); and served as a committee member for Mrs. Nichole Schimanski (PhD, 2014). Ristenpart advised Ms. Alexis Fisher (MS 2013) and Ms. Chih-Ching Chen (PhD). Both PIs are advising or already graduated several other students, now working at Google, Tektronics, Facebook, Sandia National Laboratories, Amazon.

PI Shrimpton is working with local academic and industry colleagues to reach potential CS students earlier in the educational pipeline. Portland State recently hosted a cybersecurity summer camp, part of an program sponsored by the Department of Homeland Security, which is scheduled to run for (at least) the next two years. Shrimpton and colleagues are currently developing plans for an outreach center that fosters relationships with local K-12 educators.

Interaction with industry. Our research, if successful, will improve security in a number of critical contexts. To facilitate that, we maintain an active network of industry contacts, which, among other things, helps us see and navigate around potential deployment roadblocks. Through his industry connections as former Chief Scientist of RSA, The Security Division of EMC, PI Juels is helping build ties more broadly between the scientific community and those interested in commercial deployment; he has ongoing research collaborations and/or deployment projects, for instance, with colleagues at Cisco, Qualcomm, Microstrategy, and Two Sigma, mainly in applied cryptography.

Technological impact and software dissemination. This project will have its success measured in large part by the degree to which the research impacts real-world artifacts. The PIs have a track record in this respect: FTE now ships as part of the Tor browser bundle and is integrated with Google Idea's uProxy product. PI Shrimpton's work on tweakable ciphers is now part of products sold by Voltage Security. PI Ristenpart's work on format-preserving encryption formed the basis for algorithms used widely in industry to protect credit card numbers and other sensitive

data. His work on privacy-preserving device-tracking led to open-source software downloaded over 100,000 times. His work finding new types of vulnerabilities in cloud computing and elsewhere has led to security fixes in a variety of products. PI Juels ran the research program at RSA and was responsible for the translation of a range of cryptographic research innovations into products.

To facilitate technology transfer and impact, we will make public and open-source the software prototypes that result from the proposed research. We are strong believers in the idea that making available software produced in the course of publicly-funded research accelerates scientific advancement, technology transfer, and education. We have a track record of doing this as well: our software implementations for FTE are open source (see `http://fteproxy.org` and `https://libfte.org`).

Developing the scientific community. An important part of our longer term work is development of the applied cryptography research community, which requires integrating better disparate disciplines within computer science. We particularly target expanding the interaction between those building and deploying systems and the cryptographic theory research communities. PIs Ristenpart and Shrimpton are both on the steering committee of the Real World Cryptography workshop, now in its fourth year; we intend to continue over the lifetime of this grant. This workshop brings together practitioners and academics to hear about the latest applied cryptography research as well as industry problems, and is already popular venue (with some 400 attendees and both practitioners and academics). We believe it is strengthening the sometimes fractious community of cryptography researchers who (want to) do more applied work.

As part of our proposal, we will develop new methodologies for assessing applied cryptography. In particular, by explicitly building into the design and formal analysis process as well as empiricism. We believe this data-driven approach will lead to better results, with theory tailored better to the problems of practical relevance. By interaction with the academic community via conferences, workshops, and university visits we will both advertise this methodological approach and gain feedback on it.

Engaging the anti-censorship community. Our previous work on FTE has lead us to significant interaction with, and impact on, the anti-censorship activist community. We have frequent discussions with tool designers such as the Tor team, Lantern, Google's uProxy, and others. PIs Ristenpart and Shrimpton are active members of this community and helped Google Ideas (a philanthropic team within Google) organize a workshop in July 2014 on DPI-resistance for anti-censorship tools. It was attended by a variety of practitioners and academics, and had explicit goal of helping build a cohesive community in addition to identifying research problems and potential approaches for them.

An explicit goal of our research is positive impact for activists and others who need secure, unfettered Internet communications. Our work on FTE was implemented in Tor and Google's uProxy. It is now included as a feature in the Tor bundle, so activists and others can use it. We will pursue similar impact with the DSC work proposed here.

7 Results of Prior NSF Support

Juels is a co-PI of NSF Grant #1330599, "TWC: Frontier: Collaborative: Rethinking Security in the Era of Cloud Computing," receiving $614,087 over the period 9/01/2014 – 8/31/2018. Ristenpart is the PI or co-PI for three current NSF grants, #1065134, "TC: Medium: Collaborative Research: Random Number Generation and Use in Virtualized Environments", for total $749,149 and with period of support 9/1/2011 – 8/31/2015; #1330599, "TWC: Frontier: Collaborative: Rethinking Security in the Era of Cloud Computing," receiving $1,995,068 over the period 9/01/2013 – 8/31/2018; #1253870, "CAREER: Infrastructure for Secure Cloud Computing", for a total of $480,620 over the period 6/1/2013 – 5/31/2018. Shrimpton is the PI of NSF Grant #0845610, "CAREER: Design Principles for Cryptographic Hash Functions: Foundations, Primitives and Transforms", for $400,000 and with period 6/2009–6/2014, and also of NSF grant #1319061 "Tweakable-blockcipher-based Cryptography", for $433,000 and with period 06/2013–07/2016. There is no substantive overlap in technical content of these proposals and the current proposal.

Intellectual Merit. These awards have resulted in a number of results across security and cryptography, including setting theoretical foundations for hashing, developing new architectures for cloud security, and understanding how to build and analyze random number generators (RNGs). The grants have funded work that resulted in many top-tier publications, including [7, 8, 18, 26, 30, 31, 35–37, 39, 49, 56, 60, 61, 66, 67, 72, 75, 81, 82, 88], and two best-paper awards.

Broader Impact. Our work has impacted a number of diverse areas. A number of the cited publications influenced the designs of several of the NIST SHA-3 entrants [7, 8, 30, 72]. Work uncovering side-channel attacks in commerical PaaS clouds [88] resulted in a CERT advisory. FTE [36, 60, 61] now ships with the Tor browser bundle and Google's uProxy. Work on tweakable ciphers [75] has been incorporated into Voltage Security's product line. We are working with Linux kernel developers on incorporating our new RNG designs [37].

Collaboration Plan

The team. We have assembled an accomplished team of three PIs with lengthy track records of successful collaborations, as well as expertise in the target domains of this proposal. The team participants are:

- **Prof. Ari Juels** works on applied cryptography and system security. His pioneering work on fuzzy cryptography bears directly on this proposal, as does his recent research on honey objects. Having spent many years in industry, most recently as Chief Scientist of RSA, The Security Division of EMC, Juels offers a valuable perspective on commercial deployment in support of the project's goal of translation to practice. He has also been an active researcher, having published over 75 scholarly, peer-reviewed research articles on security and cryptography.
- **Prof. Thomas Ristenpart** works in systems security and both applied and theoretical cryptography, focusing on solving real-world problems with theoretically sound cryptographic tools. His work has lead to standards used widely in industry [5]; research prototypes downloaded >100,000 times [71]; discovery of security vulnerabilities in the cloud [73, 80, 88, 89], embedded systems [22, 41], the use of machine learning [40], and in cryptographic protocols [18,26,67]; and first-of-their-kind empirical measurement studies of the cloud [45,82]. He has published 42 peer-reviewed papers on security and cryptography, mostly in top tier venues and including one best paper award.
- **Prof. Thomas Shrimpton** is an expert on practice-motivated and provably-secure cryptography, particularly in the areas of authenticated encryption, hash functions and symmetric primitives. Prior to cryptography, Shrimpton made contributions in signal processing, signal detection, and communication theory. Most recently, he has worked on cryptographic systems for censorship-circumvention. He has published 27 peer-reviewed articles, nearly all at top-tier venues, one winning a best paper award.

As can be seen, the PIs cover all core areas of expertise in the proposed research, with extensive experience in conducting empirical measurement studies, forging new cryptographic primitives, developing cryptographic theory, and building practical systems. Should the need arise for other areas of expertise, we will seek out appropriate collaborators; the PIs have a strong track record of cross-disciplinary collaboration.

Collaboration history. The three PIs have an extensive history of collaboration. Juels and Ristenpart have published five papers together in top security and cryptography venues over the past few years [38, 54, 55, 88, 89], including two on honey encryption. Although in a different technical domain than this proposal, they are co-PIs on the NSF Frontier grant "TWC: Frontier: Collaborative: Rethinking Security in the Era of Cloud Computing." Ristenpart and Shrimpton have published nine papers together in top tier security and cryptography venues, including on topics related to this proposal, such as censorship circumvention [35, 36, 60, 61].

Team coordination. The three PIs in this proposal have a longstanding practice of collaboratively advising graduate students. Juels and Ristenpart have served on the dissertation committee of one student, Yinqian Zhang (UNC), whom they collaboratively advised over several years alongside his formal advisor (Prof. Michael Reiter). Juels is already informally advising one of Ristenpart's student Rahul Chatterjee on ongoing work on SweetPass (see Section 2 in the main proposal). Ristenpart has informally co-advised Shrimpton's student Kevin Dyer on anti-censorship topics [35, 36, 61]. Ristenpart frequently co-advises students with colleagues at Wisconsin in different areas such as operating systems (Prof. Michael Swift), networking (Prof. Aditya Akella), and programming languages (Prof. Somesh Jha).

We have found close joint advising of students to be an effective foundational means of both tightly coordinating activities and exposing students to a diverse range of research styles and perspectives. We will continue to use collaborative advising as a basic vehicle to advance and coordinate our research.

Building on our various strong collaborative foundations, we will build a three-way team that achieves tight communication, a sense of ownership, and overall team cohesion in the following ways:

- **Videoconferences:** Juels and Ristenpart and Ristenpart and Shrimpton have a long history of regular, joint videoconferences with students, as well as a weekly one-on-one videoconferences to coordinate their research. We will supplement these longstanding meetings with bi-weekly, three-way group meetings via videoconferences for brainstorming, idea development, and execution. The frequency of meetings around specific projects will be determined according to technical need, with biweekly meetings by default.
- **Visits:** The three PIs have pairwise face-to-face meetings several times a year in visits to one another's institutions and at major security and cryptography conferences (e.g., ACM CCS, CRYPTO, USENIX Security).

Additionally, Ristenpart and Shrimpton have recently instituted the practice of sending graduate students to visit one another's institutions. We will extend these practices into a three-way program of physical meetings between and among PIs on at least a quarterly basis. For the purposes of student enrichment and enhanced coordination among the participating universities, we will also engage in brief cross-institution student exchanges among Cornell Tech, PSU, and Wisconsin during the academic year as well as month-long exchanges during the summers.

- **Joint industry engagement:** As tight integration of empiricism with theory and translation to practice are key elements of our proposal, we will leverage collaborative engagement with industry partners in the practice-oriented end of our work. As one example, Juels and Ristenpart are serving on the cryptography advisory board of Skyhigh Networks, where they work together with academic colleagues to translate research results, such as FTE, into advances in commercial practice. As another, Shrimpton has regular meetings with security researchers at Intel, whose group is located just a few miles from his campus. Ristenpart and Shrimpton were asked by Google Ideas to organize a workshop on obfuscation techniques, in theory and practice; their collaboration with Google Ideas is ongoing. We will seek out further such opportunities as conduits for the results of our proposal and as means of coordinating its translational aspects.

We will additionally make use of standard tools, including e-mail, SVN, and Git to ensure timely and well documented coordination of our research and code development.

Roles, assignments, and timeline

In the spirit of methodological enrichment, at least two PIs will be involved in each facet of the proposed work. To ensure against diffusion of responsibility, however, each PI will serve as a "supervisor" on certain portions of the project. The goal of the supervisor is to take primary responsibility for progress on the given task and to ensure that any potential hurdles are overcome. Table 1 presents a list of tasks, a rough timeline, and supervisory roles. Note that multiple students/PIs will be working on a task simultaneously. We have organized the work across the various themes of HE, Empirical work, DSE, Password-based Steganography, DSSS, and the Unifying Framework.

Topic	Task	Supervisor	Timeline Years
HE	(**1**) Improved DTEs	Juels	1
HE	(**2**) HE security notions	Ristenpart	1
HE	(**3**) Models and practical HE constructions	Juels	1–2
Empirical	(**4**) Trace collection infrastructure	Ristenpart	1
Empirical	(**5**) Analysis methods / attacks	Shrimpton	1–4
DSE	(**6**) Foundations of generalized DTEs	Ristenpart	1–3
DSE	(**7**) Generalized DTEs for censorship apps	Shrimpton	1–3
DSE	(**8**) Tunneling DTEs	Juels	2–4
DSE	(**9**) Stateful schemes	Shrimpton	2–4
Password-based steganography	(**10**) Formal definitions	Ristenpart	2–3
Password-based steganography	(**11**) Analyzing Hopper et al.	Shrimpton	2
Password-based steganography	(**12**) New constructions	Juels	2–4
DSSS	(**13**) Formal definitions	Ristenpart	2–3
DSSS	(**14**) Password-focused tool development	Juels	2–4
Unifying framework	(**15**) Framework development	Shrimpton	1–4
Unifying framework	(**16**) libdsc development	Ristenpart	3–4

Table 1: Tasks, supervisory roles of different PIs, and approximate timeline.

Data Management Plan

Our proposed experiments, system development, and outreach activities will produce a number of different data artifacts. As dissemination of these artifacts is essental to the goal of broad impact that is central to this proposal, we have carefully formulated distinct data management policies according to artifact types and sensitivies. We anticipate that our proposal will result broadly in the following three data types:

(1) Data sets resulting from measurement studies and experiments;

(2) Open-source software; and

(3) Curriculum materials.

Data stewards. Each task supervisor will be a designated steward for all data resulting from the task, assuming responsibility for the management of the data and determining an appropriate classification (public, conditionally sharable, or private) for particular data artifacts. Should a data steward be unable to assume continuing responsibility for certain datasets, due to departure from his institution or some other event, he will transfer stewardship to another PI on this project or to a senior researcher at his university and will furnish the instruction and documention required for the new steward to assume continuing responsibility.

Data-handling policies. We will make our data artifacts available to other researchers as well as the general public to the greatest possible extent, as consistent with privacy considerations. Our approach will be to identify datasets as *public* (P), *conditionally releasable* (S), *confidential* (C), or *educational* (E). We specify the associated policies for each below. We will apply policy (P), (S), (C), or (E) to category (1) data as deemed appropriate by the associated data steward. We will apply policy (P) to category (2) data and policy (E) to category (3) data by default. Our data-handling policies are as follows:

- *Policy (P): Public data.* Public datasets will be those suitable for posting online, e.g., data derived from public sources, or the outputs of experiments (e.g., data, source code) that themselves do not involve any privacy-sensitive data. Our policy will be to retain these data for five years from the date of publication of any paper relying on the data. We will retain data for a longer period of time if possible, giving explicit priority to the goal of ensuring long-term scientific reproducibility. Public data will be made available via a project website or a public cloud. Larger data sets that cannot be disseminated by either such means will be stored locally and instructions will be published for interested researchers and others to obtain access to the data. We will adhere to a policy of releasing all source code resulting from the proposal as open-source software under suitable nonrestrictive licenses, and will make use of repositories, e.g., GitHub, that support this practice.

- *Policy (S): Conditionally releasable data.* Some data artifacted produced by our work will carry either temporary sharing limitations (e.g., individually requested moratoria on the release of personal data) or permanent ones. We will retain such data for the same duration of time as specified in policy (P). These data will not be made public, but stored locally with appropriate access-control mechanisms to restrict both external and internal access or in a cloud with protections that are suitable to the sensitivity of the data, e.g., a HIPAA-compliant cloud. Should researchers or others submit appropriate requests for data access, we will confirm that the request is appropriate (e.g., under the aegis of IRB-approved work) and will determine a practicable minimal-release strategy, specifically exploring time-limited and sanitized data-sharing approaches, as well as whether data should be released directly or through a query interface. We will release the data as expeditiously as possible, consistent with resource and policy constraints.

- *Policy (C): Confidential data.* A data steward may deem some data temporarily or permanently unsuitable for release outside his institution. University network packet traces such as we expect to collect in the course of this proposal will be deemed confidential in all cases, while derived data such as non-personally identifiable aggregate statistics or anonymized packet headers may be categorized as (S) or (P). Other data may additionally be deemed confidential by the data steward. At the time of data collection, the steward will determine whether it is appropriate to erase the data. (For example, highly sensitive data not employed in research may be summarily deleted.) Otherwise, the data will be preserved according to Policy (S), but with no access granted outside the institution of the data steward.

- *Policy (E): Educational data.* The data produced in curriculum development in the context of this project will be handled under Policy (P). These data will be made publicly accessible on the website of the data steward or in an appropriately locatable and accessible public archive.

Data storage and lifetime. The volume of data produced in this proposal will be small enough to permit handling within the existing data storage facilities of our respective universities. At a minimum, data will be stored for the duration of the project. We anticipate storing most data for a considerably more extended period of time, however, and will store for as long as is practical both data required to reproduce published experiments and data of public value. We will store all data in suitable standard formats and will confirm that university facilities include access controls and encryption as suitable for the handling of specific data artifacts.

Vulnerability disclosures. This research project does not explicitly encompass vulnerability assessments. It is very well possible, however, that we will discover security vulnerabilities or inapproprate data disclosures in the course of our work. For example, in our statistical modeling of data we may uncover inadvertent leakage of personally identifiable information; in our study of censorship circumvention we may discover vulnerabilities that expose confidential data to censors.

We will adhere broadly to community-standard responsible disclosure practices. Specifically, we will follow the following steps in disclosing a vulnerability:

1. *We will identify stakeholders.* We will identify primary stakeholders, entities developing or managing the affected systems or data, as well as secondary stakeholders, those potentially harmed by the vulnerability, e.g., users of the impacted system or subjects of the relevant data. We will work as advocates for secondary stakeholders throughout the disclosure process.

2. *We will privately disclose the vulnerability.* We will notify primary stakeholders of the vulnerability and provide tangible evidence so that they can confirm and assess its scope. We will seek to make this disclosure as expeditiously as possible.

3. *We will assist in vulnerability remediation.* We will advise primary stakeholders on technical remediation strategies, as appropriate.

4. *We will create a public disclosure plan.* In consonance with research community practice, we will by default make a public disclosure that specifically identifies the vulnerability, modifying this approach if it may bring about harm to secondary stakeholders and working with primary stakeholders to determine the appropriate level of detail to disclose about the vulnerability. Upon discovery of the vulnerability, we will set a target date for public disclosure. By default, this will be 90 days from private disclosure of the vulnerability.

5. *We will conduct a review with primary stakeholders.* We will circulate drafts of the public disclosure to primary stakeholders, soliciting their feedback and working with them to ensure that details are correct and amending the disclosure as appropriate, taking into account any harm that may affect primary and secondary stakeholders as a result of disclosure.

6. *We will make a public disclosure of the vulnerability.* We will publish the disclosure, including both technical detail and explanations accessible to secondary stakeholders, as warranted by the vulnerability.

References Cited – Proposal Section (e)

References

[1] Anonymous. Torproject.org blocked by GFW in china: Sooner or later? Available at: https://blog.torproject.org/blog/torprojectorg-blocked-gfw-china- sooner-or-later, June 2008.

[2] Jacob Appelbaum and Nick Mathewson. Pluggable transports for circumvention. Available at: https://gitweb.torproject.org/torspec.git/HEAD:/proposals/180-pluggable-transport.txt, 2010.

[3] Michael Backes and Christian Cachin. Public-key steganography with active attacks. Technical Report 2003/231, Cryptology e-print archive, http://eprint.iacr.org, 2004.

[4] M. Bellare, T. Ristenpart, and S. Tessaro. Multi-instance security and its application to password-based cryptography. In *Advances in Cryptology – CRYPTO 2012*, pages 312–329. Springer Berlin Heidelberg, 2012.

[5] Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. Format-preserving encryption. In Michael Jacobson, Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *Selected Areas in Cryptography*, volume 5867 of *Lecture Notes in Computer Science*, pages 295–312. Springer Berlin / Heidelberg, 2009.

[6] Piotr Berman and Martin Fürer. Approximating maximum independent set in bounded degree graphs. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 365–371, 1994.

[7] John Black, Martin Cochran, and Thomas Shrimpton. On the impossibility of highly-efficient blockcipher-based hash functions. *J. Cryptology*, 22(3):311–329, 2009.

[8] John Black, Phillip Rogaway, Thomas Shrimpton, and Martijn Stam. An analysis of the blockcipher-based hash functions from PGV. *Journal of Cryptology*, 23(4):519–545, 2010.

[9] Blue coat technical brief: Policy best practices. Available at: https://bto.bluecoat.com/sites/default/files/tech_briefs/Policy_Best_Practices.1.pdf.

[10] J. Bonneau. *Guessing human-chosen secrets*. PhD thesis, University of Cambridge, May 2012.

[11] J. Bonneau. The science of guessing: analyzing an anonymized corpus of 70 million passwords. In *IEEE Symposium on Security and Privacy*, pages 538–552, 2012.

[12] Xavier Boyen. Reusable cryptographic fuzzy extractors. In *ACM Conference on Computer and Communication Security*, pages 82–91, 2004.

[13] Jon Brodkin. Iran reportedly blocking encrypted internet traffic. Available at: http://arstechnica.com/tech-policy/2012/02/iran-reportedly-blocking-encrypted-internet-traffic/, 2012.

[14] Christian Cachin. An information-theoretic model for steganography. In *Information Hiding*, volume 1525 of *Lecture Notes in Computer Science*, pages 306–318. Springer Berlin / Heidelberg, 1998.

[15] Ran Canetti, Ling Cheung, Dilsun Kaynar, Moses Liskov, Nancy Lynch, Olivier Pereira, and Roberto Segala. Task-Structured Probabilistic I/O Automata. In *International Workshop on Discrete Event Systems (WODES'06)*, pages 207–214. IEEE, 2006.

[16] Rajarathnam Chandramouli, Mehdi Kharrazi, and Nasir Memon. Image steganography and steganalysis: Concepts and practice. In *Digital Watermarking*, pages 35–49. Springer, 2004.

[17] R. Chatterjee, M. Doescher, J. Bonneau, A. Juels, and T. Ristenpart. Sweetpass: Honey encryption for password vaults. In submission, 2014.

[18] Stephen Checkoway, Matthew Fredrikson, Ruben Niederhagen, Matthew Green, Tanja Lange, Thomas Ristenpart, Daniel J Bernstein, Jake Maskiewicz, and Hovav Shacham. On the Practical Exploitability of Dual EC DRBG in TLS Implementations. In *USENIX Security Symposium*, pages 319–335. USENIX, 2014.

[19] Abbas Cheddad, Joan Condell, Kevin Curran, and Paul Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal Processing*, 90(3):727–752, 2010.

[20] Richard Clayton, Steven J. Murdoch, and Robert N. M. Watson. Ignoring the great firewall of china. In *in G. Danezis & P. Golle, eds, Privacy Enhancing Technologies workshop (PET 2006), LNCS*. Springer-Verlag, 2006.

[21] Jeddiah R. Crandall, Daniel Zinn, Michael Byrd, Earl T. Barr, and Rich East. Conceptdoppler: A weather tracker for internet censorship. In *ACM Conference on Computer and Communications Security*, CCS '07, pages 352–365. ACM, 2007.

[22] Drew Davidson, Benjamin Moench, Thomas Ristenpart, and Somesh Jha. Fie on firmware: Finding vulnerabilities in embedded systems using symbolic execution. In *USENIX Security*, pages 463–478, 2013.

[23] Rachna Dhamija and J. Doug Tygar. The battle against phishing: Dynamic security skins. In *Symposium on Usable Privacy and Security (SOUPS)*, pages 77–88. ACM, 2005.

[24] Roger Dingledine. Iran blocks Tor; Tor releases same-day fix. Available at: `https://blog.torproject.org/blog/iran-blocks-tor-tor-releases-same-day-fix`, 2011.

[25] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *In Proceedings of the 13th USENIX Security Symposium*, pages 303–320, 2004.

[26] Y. Dodis, T. Ristenpart, J. Steinberger, and S. Tessaro. To Hash or Not to Hash Again? (In)differentiability Results for H^2 and HMAC. In *Advances in Cryptology – CRYPTO '12*, pages 348–366. Springer, 2012.

[27] Yevgeniy Dodis, Jonathan Katz, Leonid Reyzin, and Adam Smith. Robust fuzzy extractors and authenticated key agreement from close secrets. In *Advances in Cryptology—CRYPTO*, pages 232–250, 2006.

[28] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. Technical Report 2003/235, Cryptology ePrint archive, `http://eprint.iacr.org`, 2006. Previous version appeared at *EUROCRYPT 2004*.

[29] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Computing*, 38(1):97–139, 2008.

[30] Yevgeniy Dodis, Thomas Ristenpart, and Thomas Shrimpton. Salvaging merkle-damgård for practical applications. In Antoine Joux, editor, *Advances in Cryptology - EUROCRYPT 2009*, volume 5479, pages 371–388, 2009.

[31] Yevgeniy Dodis, Thomas Ristenpart, and Salil P. Vadhan. Randomness condensers for efficiently samplable, seed-dependent sources. In Ronald Cramer, editor, *Theory of Cryptography Conference (TCC)*, volume 7194 of *Lecture Notes in Computer Science*, pages 618–635. Springer, 2012.

[32] Peter Dorfinger, Georg Panholzer, and Wolfgang John. Entropy estimation for real-time encrypted traffic identification (short paper). In Jordi Domingo-Pascual, Yuval Shavitt, and Steve Uhlig, editors, *Traffic Monitoring and Analysis*, volume 6613 of *Lecture Notes in Computer Science*, pages 164–171. Springer Berlin Heidelberg, 2011.

[33] Peter Dorfinger, Georg Panholzer, Brian Trammell, and Teresa Pepe. Entropy-based traffic filtering to support real-time skype detection. In *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, IWCMC '10, pages 747–751, New York, NY, USA, 2010. ACM.

[34] Holger Dreger, Anja Feldmann, Vern Paxson, and Robin Sommer. Operational experiences with high-volume network intrusion detection. In *ACM Conference on Computer and Communications Security*, pages 2–11. ACM, 2004.

[35] Kevin P Dyer, Scott E Coull, Thomas Ristenpart, and Thomas Shrimpton. Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 332–346. IEEE, 2012.

[36] Kevin P. Dyer, Scott E. Coull, Thomas Ristenpart, and Thomas Shrimpton. Protocol misidentification made easy with format-transforming encryption. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security*, CCS '13, pages 61–72, New York, NY, USA, 2013. ACM.

[37] Adam Everspaugh, Yan Zhai, Robert Jellinek, Thomas Ristenpart, and Michael Swift. Not-So-Random Numbers in Virtualized Linux and the Whirlwind RNG. In *IEEE Symposium on Security & Privacy*, pages 559–574. IEEE, 2014.

[38] Ben Farley, Venkatanathan Varadarajan, Kevin Bowers, Ari Juels, Thomas Ristenpart, and Michael M. Swift. More for Your Money: Exploiting Performance Heterogeneity in Public Clouds. In *ACM Symposium on Cloud Computing – SOCC '12*. ACM, 2012.

[39] M. Fischlin, A. Lehmann, T. Ristenpart, T. Shrimpton, M. Stam, and S. Tessaro. Random oracles with(out) programmability. In *Advances in Cryptology – ASIACRYPT 2010*, pages 303–320. Springer Berlin Heidelberg, 2010.

[40] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *USENIX Security Symposium*, pages 17–32. USENIX, 2014.

[41] W. Frisby, B. Moench, B. Recht, and T. Ristenpart. Security Analysis of Smartphone Point-of-Sale Systems. In *Workshop on Offensive Technologies – WOOT*. USENIX Association, 2012.

[42] Niklas Frykholm and Ari Juels. Error-tolerant password recovery. In *ACM Conference on Computer and Communications Security*, pages 1–8, 2001.

[43] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 40–49. IEEE, 2013.

[44] Magnús M. Halldórsson. Approximations of weighted independent set and hereditary subset problems. *J. Graph Algorithms and Applications*, 4(1):1–16, 2000.

[45] Keqiang He, Alexis Fisher, Liang Wang, Aaron Gember, Aditya Akella, and Thomas Ristenpart. Next stop, the cloud: Understanding modern web service deployment in ec2 and azure. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 177–190. ACM, 2013.

[46] Nicholas Hopper, John Langford, and Luis von Ahn. Provably secure steganography. In Moti Yung, editor, *Advances in Cryptology – CRYPTO 2002*, volume 2442 of *Lecture Notes in Computer Science*, pages 119–123. Springer Berlin / Heidelberg, 2002.

[47] Amir Houmansadr, Chad Brubaker, and Vitaly Shmatikov. The parrot is dead: Observing unobservable network communications. In *IEEE Symposium on Security and Privacy*, pages 65–79. IEEE Computer Society, 2013.

[48] Z. Huang, E. Ayday, J.-P. Hubaux, J. Fellay, and A. Juels. Genoguard: Protecting genomic data via honey encryption. In submission, 2014.

[49] Robert Jellinek, Yan Zhai, Thomas Ristenpart, and Michael Swift. A day late and a dollar short: The case for research on cloud billing systems. In *HotCloud*. ACM, 2014.

[50] Neil F Johnson and Sushil Jajodia. Exploring steganography: Seeing the unseen. *Computer*, 31(2):26–34, 1998.

[51] A. Juels and R. Rivest. Honeywords: Making password-cracking detectable. In *ACM Conference on Computer and Communications Security – CCS 2013*, pages 145–160. ACM, 2013.

[52] A. Juels and M. Sudan. A fuzzy vault scheme. *Designs, Codes and Cryptography*, 38(2):237–257, 2006.

[53] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *ACM Conference on Computer and Communications Security*, pages 28–36. ACM Press, 1999.

[54] Ari Juels and Thomas Ristenpart. Honey encryption: Encryption beyond the brute-force barrier. *IEEE Security and Privacy*, 12(4):59–62, 2014.

[55] Ari Juels and Thomas Ristenpart. Honey Encryption: Security Beyond the Brute-Force Bound. In *Advances in Cryptology–EUROCRYPT 2014*, pages 293–310. Springer Berlin Heidelberg, 2014.

[56] Will Landecker, Thomas Shrimpton, and R. Terashima. Tweakable blockciphers with beyond birthday-bound security. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology–CRYPTO '12*, volume 7417 of *Lecture Notes in Computer Science*, pages 14–30. Springer Berlin / Heidelberg, 2012.

[57] Andrew Lewman. China block Tor: Round Two. Available at: `https://blog.torproject.org/blog/china-blocking-tor-round-two`, 2010.

[58] Andrew Lewman. Update on Internet Censorship in Iran. Available at: `https://blog.torproject.org/blog/update-internet-censorship-iran`, 2011.

[59] Zhiwei Li, Warren He, Devdatta Akhawe, and Dawn Song. The emperor's new password manager: Security analysis of web-based password managers. In *23rd USENIX Security Symposium (USENIX Security 14)*, 2014.

[60] Daniel Luchaup, Kevin P. Dyer, Somesh Jha, Thomas Ristenpart, and Thomas Shrimpton. Libfte: a toolkit for constructing practical, format-abiding encryption schemes. In *Proceedings of the 23rd USENIX conference on Security Symposium*, pages 877–891. USENIX Association, 2014.

[61] Daniel Luchaup, Thomas Shrimpton, Thomas Ristenpart, and Somesh Jha. Formatted enryption beyond regular languages. In *ACM Conference on Computer and Communications Security*. ACM, 2014.

[62] Anna Lysyanskaya and Mira Meyerovich. Provably secure steganography with imperfect sampling. In *Public Key Cryptography-PKC 2006*, pages 123–139. Springer, 2006.

[63] Jack McCarthy. China bans skype. Available at: `http://www.infoworld.com/t/business/china-bans-skype-911`, 2005.

[64] H. Mohajeri Moghaddam, B. Li, M. Derakhshani, and I. Goldberg. Skypemorph: protocol obfuscation for tor bridges. In *Proceedings of the 2012 ACM conference on Computer and Communications Security*, pages 97–108. ACM, 2012.

[65] Sándor Molnár and Marcell Perényi. On the identification and analysis of skype traffic. *International Journal of Communication Systems*, 24(1):94–117, 2011.

[66] Onur Özen, Thomas Shrimpton, and Martijn Stam. Attacking the Knudsen-Preneel compression functions. In Seokhie Hong and Tetsu Iwata, editors, *Fast Software Encryption*, volume 6147, pages 94–115, 2010.

[67] Kenneth Paterson, Thomas Ristenpart, and Thomas Shrimpton. Tag size does matter: Attacks and proofs for the TLS record protocol. In *Advances in Cryptology–ASIACRYPT 2011*, LNCS, pages 372–389. Springer, 2011.

[68] Vern Paxson. Bro: a System for Detecting Network Intruders in Real-Time. *Computer Networks*, 31(23-24):2435–2463, 1999.

[69] PKCS #5: Password-based cryptography standard (rfc 2898). RSA Data Security, Inc., September 2000. Version 2.0.

[70] Niels Provos and Peter Honeyman. Hide and seek: An introduction to steganography. *Security & Privacy, IEEE*, 1(3):32–44, 2003.

[71] Thomas Ristenpart, Gabriel Maganis, Arvind Krishnamurthy, and Tadayoshi Kohno. Privacy-preserving location tracking of lost or stolen devices: Cryptographic techniques and replacing trusted third parties with dhts. In *USENIX Security Symposium*, pages 275–290. USENIX Association, 2008.

[72] Thomas Ristenpart, Hovav Shacham, and Thomas Shrimpton. Careful with composition: Limitations of the indifferentiability framework. In Kenneth G. Paterson, editor, *Advances in Cryptology–EUROCRYPT '11*, volume 6632 of *Lecture Notes in Computer Science*, pages 487–506, 2011.

[73] Thomas Ristenpart, Eran Tromer, Hovav Shacham, and Stefan Savage. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In *ACM Conference on Computer and Communication Security Conference on Computer and Communications Security Conference on Computer and Communications Security*, pages 199–212. ACM, 2009.

[74] Oscar Santolalla. Why and how to block skype. Helsinki University of Technology, TKK T-110.5190.

[75] Thomas Shrimpton and R. Seth Terashima. Efficient and beyond-birthday-bound secure tweakable ciphers. 2012. In submission to EUROCRYPT 2013.

[76] M. Siegler. One of the 32 million with a RockYou account? you may want to change all your passwords. like now. *TechCrunch*, 14 Dec. 2009.

[77] Skype. Is skype blocked in the united arab emirates (uae)? Available at: `https://support.skype.com/en/faq/FA391/is-skype-blocked-in-the-united-arab-emirates-uae`.

[78] Mariëlle Stoelinga. An introduction to probabilistic automata. *Bulletin of the European Association for Theoretical Computer Science*, 78:176–198, 2002.

[79] Jr. Tom Zeller. The internet black hole that is north korea. *The New York Times*, 23 October 2006.

[80] Venkatanathan Varadarajan, Thawan Kooburat, Benjamin Farley, Thomas Ristenpart, and Michael Swift. Resource-freeing attacks: Improve your cloud performance (at your neighbor's expense). In *ACM Conference on Computer and Communications Security – CCS 2012*. ACM, 2012. to appear.

[81] Venkatanathan Varadarajan, Thomas Ristenpart, and Michael Swift. Scheduler-based defenses against cross-vm side-channels. In *USENIX Security Symposium*, number 687–702. USENIX, 2014.

[82] Liang Wang, Antonio Nappa, Juan Caballero, Thomas Ristenpart, and Aditya Akella. Whowas: A platform for measuring web deployments on iaas clouds. In *Internet Measurement Conference*. ACM, 2014.

[83] Zachary Weinberg, Jeffrey Wang, Vinod Yegneswaran, Linda Briesemeister, Steven Cheung, Frank Wang, and Dan Boneh. Stegotorus: a camouflage proxy for the tor anonymity system. In *ACM Conference on Computer and Communications Security*, pages 109–120, 2012.

[84] Andrew M. White, Srinivas Krishnan, Michael Bailey, Fabian Monrose, and Phillip A. Porras. Clear and present data: Opaque traffic and its security implications for the future. In *Network and Distributed System Security Symposium (NDSS)*. The Internet Society, 2013.

[85] Lance Whitney. Lastpass ceo reveals details on security breach. *CNet*, May 2011.

[86] Tim Wilde. Knock Knock Knockin' on Bridges' Doors. Available at: `https://blog.torproject.org/blog/knock-knock-knockin-bridges-doors`, 2012.

[87] Philipp Winter and Stefan Lindskog. How the Great Firewall of China is Blocking Tor. In *Free and Open Communications on the Internet*, Bellevue, WA, USA, 2012. USENIX Association.

[88] Yinqian Zhang, Ari Juels, Michael Reiter, and Thomas Ristenpart. Cross-tenant side-channel attacks in PaaS clouds. In *ACM Conference on Computer and Communications Security*, pages 990–1003, 2014.

[89] Yinqian Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Cross-vm side channels and their use to extract private keys. In *ACM Conference on Computer and Communication Security Conference on Computer and Communications Security Conference on Computer and Communications Security*, pages 305–316. ACM, 2012.