Can the Government Compel Decryption? Don't Trust — Verify

Aloni Cohen
aloni@uchicago.edu
University of Chicago
Chicago, IL, USA

Sarah Scheffler
sscheff@princeton.edu
Princeton University
Princeton, NJ, USA

Mayank Varia
varia@bu.edu
Boston University
Boston, MA, USA

ABSTRACT

If a court knows that a respondent knows the password to a device, can the court compel the respondent to enter that password into the device? In this work, we propose a new approach to the foregone conclusion doctrine from *Fisher v. U.S.* that governs the answer to this question. The Holy Grail of this line of work would be a framework for reasoning about whether the testimony implicit in any action is already known to the government. In this paper we attempt something narrower. We introduce a framework for specifying actions for which all implicit testimony is, constructively, a foregone conclusion. Our approach is centered around placing the burden of proof on the government to *demonstrate* that it is not "rely[ing] on the truthtelling" of the respondent.

Building on original legal analysis and using precise computer science formalisms, we propose *demonstrability* as a new central concept for describing compelled acts. We additionally provide a language for whether a compelled action meaningfully *entails* the respondent to perform in a manner that is "as good as" the government's desired goal. Then, we apply our definitions to analyze the compellability of several cryptographic primitives including decryption, multifactor authentication, commitment schemes, and hash functions. In particular, our framework reaches a novel conclusion about compelled decryption in the setting that the encryption scheme is deniable: the government can compel but the respondent is free to use any password of her choice.

1 INTRODUCTION

In a criminal case, if the government wants to read the contents of your encrypted phone or laptop, can they compel you to enter or disclose your password? If so, when? The question sounds simple, but it doesn't have an answer. Instead, it has too many answers.

In the past few years, several state supreme courts in the United States have grappled with this question and reached wildly different rulings. The Massachusetts Supreme Judicial Court found that you can be compelled to enter your password to decrypt the hard drive contents [9, 10], whereas the Indiana Supreme Court reached the opposite conclusion [35]. The New Jersey Supreme Court ruled that you can even be forced to disclose your password [36], whereas the Pennsylvania Supreme Court ruled that passwords themselves cannot be compelled [8]. Conflicting decisions also exist among federal circuit courts [19, 40] and district courts (e.g., [20, 34, 44, 47]).

The confusion stems from disagreement about how to adapt the law of pre-digital *act of production* cases, where the government compels production of physical documents (see e.g., [13, 43, 46]). How should we overcome this physical-digital divide? Most prior work starts from the notion that encryption isn't fully understood within the law, and reasons about how encryption technology is analogous to—or fundamentally different from—physical acts of production. This article pursues a different path: we use the fact pattern of compelled decryption as a means to interrogate our understanding of the act of production doctrine generally.

The foregone conclusion doctrine. This work exclusively considers the law in the United States, where the government's ability to compel an action is substantially limited by the Fifth Amendment to the U.S. Constitution [39]. The Fifth Amendment prohibits the government from compelling an individual to provide self-incriminating *testimony*, such as oral or written statements that "disclose the contents of his own mind" [11].

Moreover, the government cannot compel an individual—who we call the *respondent*—to perform an action which, by its very performance, would reveal *implicit testimony*. But there's a catch: an action that communicates implicit testimony can be compelled if all the implicit testimony is already a "foregone conclusion." If the government knows it, the implicit testimony would "add[] little or nothing to the sum total of the Government's information" about the respondent's mind [13], and the government would in no way be "relying on the truthtelling" of the respondent. This idea is called the *foregone conclusion doctrine* and stems from the 1976 Supreme Court case *Fisher v. United States* [13].

Complementing the law with computer science. This work pursues a formal (in the computer science sense) investigation of the foregone conclusion doctrine. Our target is a specification of the relevant law, written in the language of math and cryptography, that captures a meaningful and coherent version of the foregone conclusion doctrine. Our objective for doing so is to to gain a deeper understanding of the foregone conclusion doctrine in order to reason clearly and consistently about compelled decryption under varying present and future fact patterns.

We don't want to mechanize the law. Law is flexible and adapts to new situations, and attempts to encode society's rules into a rigid computer algorithm (aka, "code as law") typically don't end well [28, 52]. Hence, our pursuit of a computer science specification is not meant to replace the law, but *to illuminate it*. Courts and legal scholars disagree wildly about compelled decryption. Some of that is due to imperfect analogies. Some is genuine disagreement as to what the law is. Some is simply bad law. In many cases, it is unclear how the various approaches relate to one another or apply to alternative scenarios.

Our goal is to present our approach in enough detail that one can apply it to unforeseen facts and get a concrete answer. Other scholars can identify both if and where they disagree with our interpretation. We make no claims that our work is "the correct interpretation" of the doctrine. We do claim that it is reasonable and coherent, that it exposes the nuance and ambiguity in the law, and that it serves as a base for others to improve in the future.

Demonstrating foregone conclusions through verification. Consider a government actor $\mathcal{G}$ and a respondent $\mathcal{R}$. The overarching

question is whether the government can compel the respondent to perform a certain *action* $\mathcal{A}$, such as producing documents or decrypting a computer. In this work, we specify $\mathcal{A}$ as an algorithm that can request inputs from the respondent's mind, like the document's location or the computer's password. If this action involves explicit testimony, then it is strictly forbidden and the concept of foregone conclusion is irrelevant.

Otherwise, to compel this action, the government must generally show two things.[1] First, the government must establish that the respondent can perform the action $\mathcal{A}$ by submitting some convincing evidence $\mathcal{E}$ to the court. Second, in compelling the respondent to perform the action, the government must not "rely on the truthtelling" of the respondent $\mathcal{R}$.

A key insight from *Fisher* is that the government does not rely on $\mathcal{R}$'s truthtelling if it has some way of *verifying* that $\mathcal{R}$ performed the required action. The government's task is typically called "authentication" in legal scholarship because when $\mathcal{A}$ is an *act of production* then it corresponds to authenticating in the evidentiary sense. We generalize the concept because authentication of evidence is a poor fit when $\mathcal{A}$ is an *act of performance* instead; see §2 for details.

Our goal is to make *Fisher*'s verification idea precise. One approach might be to require that the government specifies a procedure $\mathcal{V}$ that checks whether the respondent $\mathcal{R}$ has performed the action *exactly*. But this approach is too restrictive. For example, suppose the action is "enter a password into this password prompt." Must $\mathcal{R}$ stand or sit? Eyes open or closed? Can $\mathcal{R}$ make a typo if it succeeds on the second try? Of course the government doesn't care about any of this – it just wants the password to be entered. Another approach might be for the government to specify a set of all acceptable actions, but this set is infinite. The upshot is that the government's interest lies in the ends, not the means, and we should approach the foregone conclusion doctrine accordingly.

Our insight is to flip $\mathcal{A}$ and $\mathcal{V}$ around. Rather than asking how to verify a particular action $\mathcal{A}$, in this work we make the verification procedure $\mathcal{V}$ the centerpiece of any foregone conclusion claim. For example, rather than compelling the respondent to "enter a password," instead the government compels the respondent to "do something that gets past the login prompt screen and to the point that we can open the file explorer." It is then up to the respondent to choose any action $\mathcal{A}$ that *conforms* to this verifier $\mathcal{V}$.

Making the verifier $\mathcal{V}$ the centerpiece of our specification immediately resolves cruel trilemma of "relying on the truthtelling." The respondent is free to perform any action it wants so long as $\mathcal{V}$ accepts; there is no longer a question of honesty or evasion. The only remaining question is how to evaluate whether the respondent's task is feasible in the first place. We impose the weakest requirement: the government must demonstrate that, based on the evidence, the respondent can perform some action $\mathcal{A}^*$ that $\mathcal{V}$ accepts. We call $\mathcal{A}^*$ the *exemplar action*, and we call $\mathcal{V}$ *demonstrable* if it has such an exemplar action. In summary, our shift in focus from $\mathcal{A}$ to $\mathcal{V}$ changes how we view compulsion: instead of "is action $\mathcal{A}$ a foregone conclusion?" we ask "is verifier $\mathcal{V}$ *demonstrable*?"

[1] These two things are neither necessary nor sufficient conditions for compelling an action. For example, they are not sufficient if the act is providing oral or written testimony, and they are not necessary if the act is not incriminating.

Meaningfulness. The cost of this shift in focus is a lack of clarity about what the government will get. Presumably the government has a target action $\mathcal{T}$ in mind that it really wants the respondent to perform. There are many demonstrable $\mathcal{V}$'s for any given scenario; for example, the trivial $\mathcal{V}$ that accepts any action is always demonstrable, but it provides no guarantee that the government will receive the result of $\mathcal{T}$ or anything remotely resembling it. Another example of a demonstrable-but-meaningless verifier is to compel the respondent to flip a coin and verify that the result is one of "heads" or "tails" (rather than, say, "42").

But sometimes, it may be possible to gather enough evidence and specify $\mathcal{V}$ in enough detail that the respondent must perform an action that is more-or-less equivalent to $\mathcal{T}$. We call this property *entailment* and we codify the claim of "more-or-less equivalent" using the cryptographic notion of extraction. That is, $\mathcal{V}$ entails $\mathcal{T}$ if the government can transform the result of any conforming action into the fruits of the action $\mathcal{T}$ it desired all along.

Our contributions. This work makes several contributions in the domains of law and computer science.

- As a matter of law, we adopt a view of implicit testimony as comprised of dual but equally important parts: ex ante *ability* and ex post *conformity*. We also present a new verification-centric approach to compelling acts while constructively satisfying the foregone conclusion doctrine (§2).
- We define the concepts of demonstrability and entailment within a formal computer science framework (§3-5). The framework is expressive enough to capture, for example, the government's uncertainty as to whether an encryption scheme is deniable or not.
- Connecting the computer science to the law, we investigate whether several acts involving cryptography are compellable within our framework (§6 and Appendix A).

Generally speaking, we believe that §2 will be of most interest to readers with a background in law, and §3-5 will appeal to a computer science audience. That said, we highlight some of the takeaways about the compellability of cryptography, for both audiences.

- Decryption by entering a password is entailable in the (typical) case that the encryption scheme is committing. This is true even when using two-factor authentication. More generally, opening *commitments* is generally entailable.
- Our framework reaches a novel middle ground conclusion if decryption is deniable – meaning that it will reveal different files when given a duress password.
- In contrast to decryption and opening commitments, the acts of encrypting and creating commitments are not entailable.
- Hash functions add a twist: compelling preimages is demonstrable and—under a cryptographic assumption—entailable.

Comparison to prior work. This article is inspired by several recent works that codify legal principles with computer science techniques [1, 4, 5, 14, 22, 29, 30], including Scheffler and Varia's related work on compelled decryption [32] that we contrast with in §7. Additionally, this work is influenced by a large body of legal scholarship on compelled decryption [6, 23–27, 31, 37, 38, 51], some of which suggest broad powers to compel decryption and some argue that the government can only compel files that they already know with "reasonable particularity."

Our framework generally aligns with the broader interpretation, and it considers all acts of performance beyond just the produced files. In the next section, we delve deeper into case law, scholarship, and the value of a verification-centric approach as a matter of law.

The question of deniable encryption highlights the differences between the current work and prior work (see Section 6.2). Deniable encryption introduces a *duress* password which, when entered, allows a device to be decrypted to something other than its true contents. In *Comm. v Gelfgatt*, the court simply ordered the respondent to use the true password rather than the duress password.[2] Sacharoff sets aside this "niche case because deniable encryption remains rare" [31]. Kerr argues that duress passwords are "unlikely to raise significant Fifth Amendment issues" because, if used, "the government will not realize what the user has done."[3] Cohen and Park argue that the government would be unable to compel decryption without some way of distinguishing the respondent's use of the true and duress passwords [6].

In contrast, our framework would allow the government to compel the respondent enter one of the true or duress passwords, but the respondent would be free to choose. If the government also had evidence as the contents of the encrypted drive—say, specific filenames as in *In re Boucher* [20]—the government could require that the apparently-decrypted device contain the known files by checking for those files in the verification procedure. In such cases, our framework yields a procedure more like Sacharoff's and Cohen-Park's than Kerr's. But like Kerr, meaningful acts are compellable even without any specific knowledge of the device's contents.

2 A NEW TAKE ON FOREGONE CONCLUSIONS

This section gives a too-brief overview of the law of the foregone conclusion doctrine and of our new approach to that doctrine. A full treatment is out of scope in this paper.[4]

The Fifth Amendment to the US Constitution provides individuals a privilege against being "compelled in any criminal case to be a witness against himself."[5] Courts and legal scholars disagree about how this privilege applies to so-called compelled decryption cases, wherein the government seeks to compel an individual to unlock or decrypt an encrypted phone, computer, or hard drive. At issue is how the foregone conclusion test applies to the act of decryption.

The Fifth Amendment's privilege applies only to acts that are "testimonial." [6] An act is testimonial when it "explicitly or implicitly[] relate[s] a factual assertion or disclose[s] information." [7] As such, a person may be compelled to furnish a blood sample [8] or to put on a shirt [9] no matter how incriminating. But a person may not be compelled to make incriminating oral or written statements.

Implicit testimony and the foregone conclusion doctrine. An act may have "testimonial aspects" even if it does not involve any oral or written testimony as such. *Fisher v. U.S.* lays out a version of this idea now called the act of production doctrine. In *Fisher*, the government issued a subpoena to compel the defendant to produce certain documents. "The act of producing the documents has communicative aspects of its own, wholly aside from the contents of the papers produced. Compliance with the subpoena tacitly concedes the existence of the papers demanded and their possession or control by the [respondent]. It also would indicate the [respondent]'s belief that the papers are those described in the subpoena." The lesson is that testimony may be implicit in an act. *Fisher*'s examples—existence, possession, and authenticity—are an application of this idea to the specific act of producing physical evidence.

Fisher also introduced the *foregone conclusion doctrine* which provides a way for the government to compel an act of production despite the implicit testimony. Kerr gives a typical account of the doctrine [24]:

> [W]hen the testimonial aspect of a compelled act "adds little or nothing to the sum total of the Government's information," any implied testimony is a "foregone conclusion" and compelling it does not violate the Fifth Amendment. . . . The Court [in *Fisher*] held that the testimony implicit in handing over the tax documents was a foregone conclusion because the government was "in no way relying on the 'truthtelling' of the [respondent]" to prove it.

In other words, "A valid privilege exists only when the compelled act is testimonial under the act of production doctrine but is not a foregone conclusion" [24]. Importantly, the foregone conclusion doctrine only considers testimony implicit in an act. It does not apply to 'pure' oral or written testimony – no matter how foregone, pure testimony cannot be compelled.[10] And the foregone conclusion doctrine is unconcerned with information that may result from the act, but is not implied by the act. Borrowing an example from Kerr, the testimony implicit in the act of opening a door to a treasure filled room does not include the treasure itself, only the "door-opening" testimony. The value of the treasure is not relevant to the Fifth Amendment privilege.

[2] *Comm. v Gelfgatt* [9] at note 10 (the decryption "protocol" states in relevant part: "The defendant shall manually enter the password or key to each respective digital storage device in sequence. . . . The defendant is expressly ordered not to enter a false or 'fake' password or key").

[3] *Kerr* [24] at note 78. It is not clear to what extent these are statements of legal doctrine (e.g., no implicit testimony at issue; the testimony is always a foregone conclusion) or of practical reality (e.g., the deniable encryption will be invisible and therefore will not be at issue in the proceedings). In one sense, Kerr agrees with us: when ordered to "bypass [the] password prompt" a respondent is free to use a duress password. But he seems to suggest that this is merely a de facto freedom, and that use of a duress password could "violate the Decryption Order." Under this reading, an order forbidding the use of the duress password (as in *Gelfgatt*, note 2 above) may be permissible even if the government "will not realize what the user has done." In contrast, this paper argues that the decryption order can only forbid the use of a duress password if the government is able to verify whether or not the true password was used. Otherwise, the act of decrypting with the true password would communicate testimony as to conformity that is not a foregone conclusion.

[4] A draft of a companion law review article was presented at the Privacy Law Scholars Workshop in 2021 [7].

[5] U.S. Const. Amend. V.

[6] Fisher v. United States, 425 U.S. 391, 408 (1976) [13]

[7] United States v Doe, 487 U.S. 201, 208 (1988) [12]

[8] Schmerber v. California, 384 U.S. 757, 763-764 (1966) [33]

[9] Holt v. United States, 218 U.S. 245 (1910) [15]

[10] *Fisher*, 258 U.S. [13] ("It is doubtful that implicitly admitting the existence and possession of the papers rises to the level of testimony within the protection of the Fifth Amendment" at 411; additionally "it [the subpoena to produce documents] does not compel oral testimony, nor would it ordinarily compel the taxpayer to restate, repeat, or affirm the truth of the contents of the documents sought[, t]herefore the Fifth Amendment would not be violated" at 409. These statements contrast this foregone conclusion doctrine against *explicit* compelled admittance of the existence and possession of the papers by oral testimony. Thus the foregone conclusion doctrine may compel only implicit testimony and not explicit oral testimony, even if that information is already known to the government.)

Ability and conformity. Applying the foregone conclusion doctrine first requires answering the question: what are the implicit testimonial aspects in the act of decryption (or some other act of performance) [6, 19, 24]? This may sound easy, but it is where many compelled decryption analyses diverge. For example, Kerr concludes that " 'I know the password' is the only assertion implicit in unlocking the device" [24]. Cohen and Park respond that "[d]eniable encryption complicates the very notion of 'the' password by introducing an alternate duress password" that appears to decrypt but hides certain files [6].

We adopt the view that there are two types of testimony implicit in acts of performance: ability and conformity. By honestly performing an act specified by the government, a respondent implicitly asserts "I can do it" and "I did do it." *Ability*—I can do it—is the respondent's ex ante belief that she is able to perform the compelled act. *Conformity*—I did do it—is the respondent's ex post belief that she did indeed perform the compelled act as specified. This view has appeared in prior works, though the terms are new. For example, Kerr describes these as the "two kinds of beliefs" implicit in a compelled act.[11]

The importance of ability to the foregone conclusion doctrine is well accepted, and the notion of ability has evolved beyond its act-of-production genesis. For physical production of some object, ability amounts to the existence of the object and the respondent's possession and control thereof, *Fisher* [13]. For entering a password, ability often amounts to the respondent's knowledge of the password [24]. For decryption and production of encrypted drives, ability communicates the respondent's "possession, control, and access to the encrypted portions of the drives; and of his capability to decrypt the files" [19].

By contrast, conformity's role is often minimized or overlooked. *Fisher* and all subsequent caselaw and scholarship focus on only one aspect of conformity: *authenticity*. In Kerr's analysis, only ability informs his analysis of compelled decryption.[12] An act of production in response to a subpoena implicitly communicates "that the articles produced are the ones demanded" [50]. But while authenticity — whether something is what it purports to be — is appropriate for acts of production, it is an ill fit for acts of *performance* where the respondent must do something instead of produce something. Compelled decryption—say, by typing a password into an encrypted computer—is typically best treated as an of performance, not production.[13] We consider conformity as a generalization of authenticity that is relevant to acts of performance as well as production, and our work gives conformity equal consideration for the first time.

A key difference between how Kerr and Cohen-Park treat compelled password entry is the importance they attach to conformity [6, 24]. Both agree on ability: the act of entering a password communicates knowledge of the password. But they disagree about conformity in the context of deniable encryption. Kerr dismisses it, stating that "the government will not realize what the user has done." Cohen and Park recognize that deniable encryption implicates conformity exactly because the government cannot tell what the user has done.

The foregone conclusion exception. Once the implicit testimonial aspects of the act of decryption are identified, the foregone conclusion doctrine asks: Are those aspects foregone conclusions [6, 19, 24]? Ability is a foregone conclusion if the government can demonstrate that the respondent can perform the action ex ante. Conformity is a foregone conclusion if the government can demonstrate ex ante that it will be able to verify whether the respondent performed the action ex post.

Ability raises no new difficulties. The government must show that the respondent can perform the act. Depending on the act, this amounts to showing that the papers exist and and are in his control [13, 46]; that he knows the password (e.g. [8–10, 34, 42, 48]); or that he is able to speak, write, or make a gesture [33]. In our formalism, the government meets its ability burden by (1) describing an *exemplar action* that is "conforming" (see below), and (2) presenting evidence that the respondent is able to perform the exemplar action.

Conformity is more challenging. Conformity is a foregone conclusion if the government can demonstrate that it will be able to *verify* whether or not the respondent performed the compelled act after the fact. Our framework requires the government to specify the procedure it will use to verify the act, and deems *any* act that successfully verifies as conforming.

Under this approach, **the verification procedure is the centerpiece of a motion to compel**. The question of conformity is resolved by construction: any action that verifies, verifies. To show that ability is a foregone conclusion, the government must demonstrate that there exists a conforming exemplar action that the respondent can perform. If so, we say the verification procedure is *demonstrable*.

Our approach to conformity addresses a shortcoming in the status quo's treatment of conformity in the guise of authenticity. In compelled decryption cases to date, courts have generally dealt with authenticity in one of four unsatisfactory ways. (In each, replacing "authenticity" with "conformity" would not fix the problem.) First, ignore the issue altogether: "the authenticity element is routinely cited but only applied loosely if at all."[14] Second, defer the question to trial, which we fear could allow the government to make improper use of fruit of a poisonous tree. Third, hold that decryption is self-authenticating—a technological assertion that deniable encryption, for example, disproves. Fourth, attempt to reason about the production of some *thing*—a password, plaintext, or ciphertext —rather than the performance of an act, a confusing and unworkable approach.[15]

Conformity should not be a mere afterthought. Even before *Fisher*, Wigmore's *Treatise on Evidence* [50] states that production of documents or chattels may only be compelled from a respondent "without the use against him of process relying on his truthtelling." Where conformity is not a foregone conclusion, the government would be improperly "relying on his truthtelling." This reliance is clear in *Comm. v. Gelgatt*, where the court simply ordered the respondent "[n]ot to enter a false or 'fake' password" although it

[11] See Kerr [24] at I.A.
[12] See Kerr [24] at II.
[13] See Cohen and Park [6], especially Sections IV.C, IV.D, V.B, V.C, and VII.
[14] In re Search of a Residence in Aptos, California 95003, No.17-mj-70656-JSC-1, 2018 WL 1400401 [21], at 10.
[15] See also Cohen and Park [6] at VII.

could not check compliance.[16] Such orders revive the cruel trilemma that is at the heart of the religious tribunals that gave rise to the right to silence: the respondent must choose among perjury, contempt of court, and self-incrimination.

A new procedure for motions to compel. Based on our framework, we suggest a new procedure for motions to compel decryption or other acts of performance. First, the government makes a motion to compel. It specifies a verification procedure and demonstrates that the respondent is able to perform some exemplar action that verifies. To do so, the government must clearly detail the steps of both the verification procedure and the exemplar action in a manner understandable by the judge and the respondent. The exemplar action must be one that the government would have been able to compel if all implicit testimony was forgone. In particular, the exemplar cannot require explicit oral or written testimony. The government must also introduce any evidence that it needs to demonstrate the respondent's ability to perform the exemplar.

Second, the respondent may challenge the motion to compel in a few different ways. She may challenge the accuracy of the government's evidence. She may also argue that she cannot perform the exemplar action. Or she may argue that the Fifth Amendment privilege precludes the exemplar action for reasons other than implicit testimony. (The respondent may *not* appeal to conformity, which is foregone by construction.)

The court must then decide whether to quash the motion or to grant it by issuing an order. The order would compel the respondent to perform an action of his choosing that passes the government-specified verification procedure. The order would contain within it a complete description of the procedure and the exemplar action.

In the new procedure based on our framework, it is important for the order to clearly specify the evidence, method of verification, and exemplar. Doing otherwise would undermine the Fifth Amendment privilege. The resulting uncertainty would present the respondent with a cruel choice: either risk contempt for failing the verification procedure or risk disclosing implicit testimony which is *not* foregone. Moreover, the idea that the government must provide the verification procedure before compelling an action is already present in, for example, *U.S. v Bright*.[17] As an added benefit, a well-defined verification procedure might be relevant in disputes over the introduction or authentication of evidence at trial time.

Under our framework, the respondent can perform *any* act that passes the government's verification procedure. Presumably the government has in mind some *target action* that it wants the respondent to perform (e.g., the exemplar). But if the government cannot tailor its verification procedure appropriately, then the compelled act may be meaningless. If however the verification is tailored so as to require something "just as good" as the target, we say that the verification procedure *entails* the target.

For example, consider the verification procedures the government could use to compel decryption of a device encrypted using deniable encryption. It could, of course, check that the device appears to be logged in. However, compelling this would clearly allow a decryption under both the true and duress passwords to pass the verification procedure. Using the duress password is allowed—the respondent would not be acting untruthfully. But if the government had specific knowledge of a file visible only using the true password, then checking for that file would prevent the duress password's use.

[16] Commonwealth v. Gelfgatt, 11 N.E.3d 605 (Mass. 2014) [9], (see footnote 10, emphasis removed)

[17] United States v. Bright, 596 F. 3d 683, 693 (9th Cir. 2010) [41] (the government does not "need to prove that it had previously authenticated the same documents … it need[s] to show only that it *could* do so" (emphasis added)).

3 COMPUTATIONAL FORMALISM

In this section, we describe the formal model of computation that we consider in this work. It is inspired by the computation and communication model within Canetti's universal composability (UC) framework [2] and it directly extends the computational model used within Scheffler and Varia's prior work on compelled actions and foregone conclusions [32, Appendix A].

Concretely, we model computation as a collection of stateful, event-driven interactive Turing machines (ITM). These Turing machines have two tapes called a method tape and an input tape. When a machine is invoked with a string Method on its method tape and an input *input* on its input tape, it executes the code of Method. Additionally, the machines have additional tapes that allow them to communicate with each other using authenticated, confidential channels, following the model of Canetti, Cohen, and Lindell [3].

In this work, we describe the methods of an interactive Turing machine using object-oriented pseudocode. Here is an example:

```
Example ITM M:
Variables: x_1, x_2 ≠ 0, x_3 ← 5
Method Set(i, x')
  | set x_i ← x'
Method Send()
  | M'.Receive(x_1, x_2, x_3)
```

Our ITMs have variables and methods. By default, all variables are private and local and all methods are public. The example machine M above has three local variables, where x_1 can be initialized arbitrarily, x_2 is initialized to some nonzero value, and x_3 is initialized to the value 5. M also has public methods M.Set and M.Send. So, we can invoke machine M by placing Set on its method tape and (i, x') on its input tape. (When an ITM has only one method and no variables, we omit naming the method.) The Send command sends a communication to another Turing machine M' that it should invoke a method of its own called Receive.[18]

Looking ahead, in this work it will be useful to model situations in which the government knows that a device has a specific behavior in some situations, but may not know the entire code of the device. For this reason, we define a partial ordering $\preccurlyeq$ on machines.

Definition 3.1. Let M_1 and M_2 be interactive Turing machines, and let S denote the set of all methods specified within M_1. We say that M_1 is a *partial specification* of M_2 (equivalently, M_2 *implements* M_1), denoted $M_1 \preccurlyeq M_2$, if:

- M_2 contains at least all of the methods specified in M_1, i.e., the methods specified in M_2 form a superset of S, and

[18] Though the UC model guides us, a complete UC specification is premature at this stage in this line of work and would make this paper inaccessible to its intended audience, most of whom are unfamiliar with the UC model. For the same reasons, we do not attempt a precise compilation from our pseudocode machines to ITMs.

- For all executions of the machines that only invoke the methods within S, if M_1 halts with some output then M_2 halts with the same output. This semantic equivalence must hold even over multiple, sequential invocations of the stateful machines M_1 and M_2.

M_1 *fully specifies* M_2, denoted $M_1 \sim M_2$ if $M_1 \preccurlyeq M_2$ and $M_2 \preccurlyeq M_1$.

In our example above, we can consider $M \preccurlyeq M''$ for any ITM M'' that contains Set and Send commands with the same code as listed above. That said, M'' might have additional commands, such as a Delete method that sets all variables to zero or a Write method that covertly overwrites a private variable. Note that $\preccurlyeq$ is uncomputable in general.

In this work, machines provided by the government (i.e., $\mathcal{V}$, $\mathcal{A}^*$, $\mathcal{T}$, and $\mathcal{P}$ defined below) tend to be fully specified. On the other hand, machines controlled by others (i.e., $\mathcal{R}$ and $\mathcal{N}$) are typically only partially specified since the government may only have some incomplete evidence about how they operate.

4 A VERIFICATION-CENTRIC APPROACH TO COMPULSION

We consider a *government* actor $\mathcal{G}$, a *respondent* $\mathcal{R}$, and the outside world (*nature*) $\mathcal{N}$. $\mathcal{N}$ is an infinite collection of ITMs $\mathcal{N}[0]$, $\mathcal{N}[1]$, $\mathcal{N}[2]$, ..., where the index is called the *location*. Oftentimes we will specify a device D as either a partial or full representation of an oracle at (for example) location ℓ in Nature, denoted as $D \preccurlyeq \mathcal{N}[\ell]$ or $D \sim \mathcal{N}[\ell]$ respectively. Some of nature's ITMs simply store information and their only functionality is a read() method that returns that information. We call such locations *read only*. In these cases, we sometimes simplify the notation by writing $x \leftarrow \mathcal{N}[1]$ instead of $x \leftarrow \mathcal{N}[1].\text{read}()$, for example.

$\mathcal{G}$ and $\mathcal{R}$ don't interact with each other or with $\mathcal{N}$ directly. Instead, they each output ITMs that specify how they choose to act. The respondent outputs the *action* $\mathcal{A}$ it will take, and the government outputs a *verifier* $\mathcal{V}$ that represents how it will authenticate the action performed by the respondent. One can think of $\mathcal{V}$ and $\mathcal{A}$ as the actions $\mathcal{G}$ and $\mathcal{R}$ choose to perform under the circumstances. Separating $\mathcal{V}$ and $\mathcal{A}$ from $\mathcal{G}$ and $\mathcal{R}$ allows us to describe and reason about the interaction between government and respondent (and nature) while assuming very little about $\mathcal{G}$ and $\mathcal{R}$.

We denote an execution $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{\mathcal{N},\mathcal{R}} \rangle$.[19] Here, $\mathcal{V}$ and $\mathcal{A}$ are ITMs that may interact freely with one another and with $\mathcal{N}$. However, $\mathcal{R}$ may only interact with $\mathcal{A}$, not with $\mathcal{N}$ nor $\mathcal{A}$. The output of the execution is $\mathcal{V}$'s output: one of Accept or Reject along with a transcript τ of its view and execution.

Definition 4.1 ($\mathcal{V}$ accepts $\mathcal{A}$)**.** We say $\mathcal{V}$ *accepts* $\mathcal{A}$ *with respect to* $\mathcal{R}, \mathcal{N}$ if $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{\mathcal{N},\mathcal{R}} \rangle$ returns Accept with probability 1 over the coins of $\mathcal{V}$, $\mathcal{A}$, $\mathcal{R}$, and machines in $\mathcal{N}$. If $\mathcal{N}$ and $\mathcal{R}$ are clear from context we abbreviate this as $\mathcal{V}$ *accepts* $\mathcal{A}$.

4.1 Evidence

We envision the government as having some *evidence* that certain states of the world ($\mathcal{N}$) or the respondent's mind ($\mathcal{R}$) are or are not true. We represent the evidence as a binary relation $\mathcal{E} : (\mathcal{R}, \mathcal{N}) \mapsto \{0, 1\}$. $\mathcal{E}(\mathcal{R}, \mathcal{N}) = 1$ means that the pair $(\mathcal{R}, \mathcal{N})$ is *consistent* with the evidence. Even if $\mathcal{E}(\mathcal{R}, \mathcal{N}) = 1$, the "true" state of the world can be some other $(\mathcal{R}', \mathcal{N}')$. Conversely, $\mathcal{E}(\mathcal{R}, \mathcal{N}) = 0$ means that $(\mathcal{R}, \mathcal{N})$ is inconsistent with the evidence. Namely, the evidence refutes that setting of $\mathcal{R}$ and $\mathcal{N}$. (Note that $\mathcal{E}$ is generally uncomputable.)

Definition 4.2 ($\mathcal{E}$-consistency)**.** Let $\mathcal{E}$ be a binary relation. We say that $\mathcal{R}, \mathcal{N}$ are $\mathcal{E}$*-consistent* if $\mathcal{E}(\mathcal{R}, \mathcal{N}) = 1$.

In our framework, $\mathcal{E}$ is exogenous and correct. We can imagine that the government has done the work to collect the evidence, introduced it in the court, and demonstrated its truth. As such, the "true" setting of $\mathcal{R}, \mathcal{N}$ is $\mathcal{E}$-consistent. (As a corollary, $\{(\mathcal{R}, \mathcal{N}) : \mathcal{E}(\mathcal{R}, \mathcal{N}) = 1\} \neq \emptyset$.) We do not consider incorrect evidence.

It will be useful to reason about cases when the government has greater or lesser evidence. We define a (weak) partial ordering over evidence which captures whether evidence $\mathcal{E}_2$ is "at least as strong as" evidence $\mathcal{E}_1$. This can arise when the government goes and gathers more evidence, or when it introduces additional evidence that it already had.

Definition 4.3 (Partial ordering of evidence)**.** Let $\mathcal{E}_1$ and $\mathcal{E}_2$ be binary relations, and let $C_{\mathcal{E}_1}, C_{\mathcal{E}_2}$ be the set of $\mathcal{E}_1$- and $\mathcal{E}_2$-consistent worlds respectively. That is, $C_{\mathcal{E}_i}$ is the set of $(\mathcal{R}, \mathcal{N})$ for which $\mathcal{E}_i(\mathcal{R}, \mathcal{N})$ outputs 1 (note that these sets may be uncomputable).

We say $\mathcal{E}_2 \succeq \mathcal{E}_1$, or $\mathcal{E}_2$ *has at least as much evidence as* $\mathcal{E}_1$, if $C_{\mathcal{E}_2} \subseteq C_{\mathcal{E}_1}$. Equivalently, $\mathcal{E}_2$-consistency implies $\mathcal{E}_1$-consistency.

As discussed in Section 3, most of the time we consider partial specifications of ITMs. In Section 6 we will sometimes find it useful to consider evidences where the machines are specified fully. We define the following useful shorthand to capture this scenario.

Definition 4.4 (Fully specified $\mathcal{E}_D$)**.** For evidence $\mathcal{E}$ and ITM D, we define $\mathcal{E}_D \succeq \mathcal{E}$ to be the evidence where every assertion of the form "$D \preccurlyeq M$" is replaced with "$D \sim M$."

4.2 Demonstrability

Our key mechanism to avoid "relying on the truthtelling" is for the government to specify a verifier $\mathcal{V}$ which will be used to verify the respondent's response. There must be some *exemplar action* $\mathcal{A}^*$ which demonstrates that the respondent is *able* to make the verifier accept, however in the context of the court case the respondent is allowed to perform *any* action $\mathcal{A}$ that results in the verifier outputting Accept.

Definition 4.5 (Demonstrability)**.** A verifier $\mathcal{V}$ is *demonstrable* with respect to evidence $\mathcal{E}$ if there exists an "efficient" (see Remark 4.6) action $\mathcal{A}^*$ such that for all $(\mathcal{R}, \mathcal{N})$ that are $\mathcal{E}$-consistent:

(1) Every method call by $\mathcal{A}^*$ to $\mathcal{R}$ produces some output.
(2) $\mathcal{V}$ accepts $\mathcal{A}^*$ with respect to $\mathcal{R}, \mathcal{N}$

The first requirement essentially requires that $\mathcal{R}$ is able to perform any actions in $\mathcal{A}^*$ that directly involve the respondent's mind. Not only does this requirement ensure that $\mathcal{E}$ contains some evidence that $\mathcal{R}$ has the method called, the fact that the method must produce some output for *all* settings of $\mathcal{R}$ ensures that $\mathcal{E}$ must specify some behavior on the output of the method (or else the $\mathcal{R}$ for which the method outputs nothing will violate the statement).

[19] We borrow this notation from oracle machines to reinforce the ways the $\mathcal{V}$, $\mathcal{A}$, $\mathcal{N}$, and $\mathcal{R}$ may and may not interact. More precisely though, $\mathcal{V}$ can invoke $\mathcal{N}$ and $\mathcal{A}$ can invoke $\mathcal{N}$ and $\mathcal{R}$ as (global) subroutines, in the UC language of Canetti [2].

Our envisioned courtroom procedure for a motion to compel would require $\mathcal{G}$ to provide $\mathcal{V}$, $\mathcal{A}^*$, and $\mathcal{E}$ to the court and to $\mathcal{R}$ (see Section §2).

Remark 4.6. We deliberately leave the "efficiency" requirement of $\mathcal{A}^*$ somewhat vague in Definition 4.5. We expect that the court will want to define "ability to perform $\mathcal{A}^*$" in a way that implies some kind of reasonable time and effort limit. For example, if $\mathcal{A}^*$ instructed $\mathcal{R}$ to decrypt a ciphertext, this is *theoretically* doable even without the key – if the respondent spends time greater than the lifetime of the universe running a brute-force decryption algorithm. Instead, we expect a more typical setting is when $\mathcal{G}$ knows $\mathcal{R}$ is capable of decrypting the ciphertext quickly (say, by using the key). We leave incorporating computational requirements to future work.

The respondent may perform any action $\mathcal{A}$ that results in the verifier outputting Accept. We call such actions *conforming*:

Definition 4.7 (Conformity)**.** *$\mathcal{A}$ conforms with $\mathcal{V}$* for a given $\mathcal{N}, \mathcal{R}$ if $\mathcal{V}$ accepts $\mathcal{A}$ with respect to $\mathcal{N}, \mathcal{R}$. If any of these algorithms are randomized, then acceptance must hold with probability 1 over all random choices. We also say $\mathcal{A}$ is *$\mathcal{V}$-conforming*.

Lemma 4.8 shows that the more evidence the government reveals, the more it can compel. An action never becomes "less compellable" by revealing more evidence. Focusing narrowly on compulsion, it is in the government's best interests to reveal all evidence it possesses. We also remark that adding more evidence only restricts the respondent's possible response actions.

Lemma 4.8 (Demonstrability, conforming are monotonic in $\mathcal{E}$)**.** Let $\mathcal{V}$ be demonstrable with respect to evidence $\mathcal{E}_1$, and let $\mathcal{A}^*$ be $\mathcal{V}$-conforming. For all $\mathcal{E}_2 \succeq \mathcal{E}_1$, $\mathcal{V}$ is demonstrable with respect to $\mathcal{E}_2$, and $\mathcal{A}^*$ is $\mathcal{V}$-conforming.

Proof. Let $C_{\mathcal{E}_i}$ be the set of $\mathcal{E}_i$-consistent $(\mathcal{R}, \mathcal{N})$ for $i \in \{1, 2\}$. By definition of demonstrability, $\forall (\mathcal{R}, \mathcal{N}) \in C_{\mathcal{E}_1}$, the execution $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{*\mathcal{N},\mathcal{R}} \rangle$ returns 1. $\mathcal{E}_2 \succeq \mathcal{E}_1$ implies that $C_{\mathcal{E}_2} \subseteq C_{\mathcal{E}_1}$. Hence, $\forall (\mathcal{R}, \mathcal{N}) \in C_{\mathcal{E}_2}$, so the execution $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{*\mathcal{N},\mathcal{R}} \rangle$ returns 1. Thus, $\mathcal{V}$ is also demonstrable with respect to $\mathcal{E}_2$, with exemplar $\mathcal{A}^*$. □

5 ENTAILMENT

In the last section, we described how, in order to avoid relying on the respondent's truthtelling, the government may provide a demonstrable verifier $\mathcal{V}$, along with an exemplar action $\mathcal{A}^*$ and evidence $\mathcal{E}$, and compel the respondent to take some action $\mathcal{A}$ that is accepted by $\mathcal{V}$. Ultimately though, the government likely has the goal of compelling the respondent to perform some specific *target action* $\mathcal{T}$. In this section, we ask: is there some demonstrable $\mathcal{V}$ that forces $\mathcal{R}$ to perform the target action $\mathcal{T}$, or something "just as good"?

To illustrate why this is a different property than demonstrability, consider that there are many demonstrable $\mathcal{V}$'s for any given scenario. If nothing else, the trivial $\mathcal{V}$ that always outputs Accept is demonstrable for any exemplar action and evidence. With this trivial $\mathcal{V}$, even though the government might hope that the respondent would choose to perform the exemplar action $\mathcal{A}^*$, the respondent could take any action $\mathcal{A}$ whatsoever in order to satisfy $\mathcal{V}$. But in some scenarios, it may be possible to specify $\mathcal{V}$ in enough detail that $\mathcal{R}$ has no choice but to perform an action that is more or less equivalent to $\mathcal{T}$.

We call this property *entailment*, and we say that $\mathcal{V}$ entails $\mathcal{T}$ if $\mathcal{G}$ can recover the fruits of $\mathcal{T}$ after $\mathcal{R}$ performs *any* $\mathcal{V}$-conforming action $\mathcal{A}$ (including but not limited to the exemplar action $\mathcal{A}^*$). Namely, the government has some method $\mathcal{P}$ that "post-processes" the result of $\mathcal{A}$ to recover what would have resulted if $\mathcal{R}$ had run $\mathcal{T}$ itself.

Definition 5.1 (Entailment)**.** Consider an oracle TM $\mathcal{T}$ (target) that takes no input and outputs a string in $\{0,1\}^*$. *$\mathcal{V}$ entails $\mathcal{T}$* with respect to evidence $\mathcal{E}$ if there exists an efficient oracle TM $\mathcal{P}$ such that for all $\mathcal{E}$-consistent $\mathcal{N}$, $\mathcal{R}$ and all $\mathcal{V}$-conforming $\mathcal{A}$:

$$\mathcal{P}^{\mathcal{N}'}(\tau) \equiv \mathcal{T}^{\mathcal{N},\mathcal{R}}(),$$

where $\tau \leftarrow \langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{\mathcal{N},\mathcal{R}} \rangle$ is the transcript of the execution between $\mathcal{V}^{\mathcal{N}}$ and $\mathcal{A}^{\mathcal{N},\mathcal{R}}$, and $\mathcal{N}'$ is the state of nature after the interaction between $\mathcal{V}$ and $\mathcal{A}$. We say $\mathcal{T}$ is *entailable* if there exists demonstrable $\mathcal{V}$ that entails $\mathcal{T}$.

To complete this definition, it only remains to define what $\equiv$ means. We begin with the easier case: if all Turing machines in the definition are deterministic, then $\equiv$ denotes exact equality. That is: the Turing machines $\mathcal{P}$ and $\mathcal{T}$ must output the same string, even if they follow very different paths to get there. The harder case is when randomness is involved (formally, when the Turing machines have randomness tapes following the model of Canetti [2]). In this case we impose a high bar for entailment: for each setting of the randomness tapes of $\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{T}, \mathcal{R}$, and all of the machines within $\mathcal{N}$, it must be the case that the outputs of $\mathcal{P}$ and $\mathcal{T}$ are identical.

Remark 5.2. The requirement that $\mathcal{P}$ is "efficient" is necessary for entailment to be meaningful. For example, consider the case of compelled decryption. If the post-processor could run for an unbounded amount of time, then $\mathcal{P}$ could simply ignore the respondent's action and brute-force the decryption, allowing the government to entail the act of decryption even when the respondent did not have the key! For this reason, Definition 5.1 requires that the post-processor $\mathcal{P}$ be efficient. As in Remark 4.6, we leave the efficiency requirement underspecified and for future work.

Remark 5.3. We note that our definition of entailment only considers target actions $\mathcal{T}$ that produce some output. As such, it does not require that the state of nature $\mathcal{N}_{A,P}$ that results after $\mathcal{A}$ and $\mathcal{P}$ is the same as the state of nature $\mathcal{N}_T$ that would have resulted if $\mathcal{R}$ performed $\mathcal{T}$. But entailment is expressive enough to capture a somewhat weaker guarantee. Suppose there is a function $q^{\mathcal{N}}$ that reads from $\mathcal{N}$ whatever the government believes ex ante is relevant, and outputs the result. If $\mathcal{V}$ entails $\mathcal{T}_q \triangleq q \circ T$, then $\mathcal{N}_{A,P}$ and $\mathcal{N}_T$ will be the same in the (ex ante) relevant part.

5.1 Impossibility of entailing unknown goals

In this section, we show that it should not be possible to entail actions where the government does not even know what property they would need to check to verify the truth. We illustrate this concept by example and then show a general theorem.

Suppose the evidence states that $\mathcal{R}$ has some secret z (for instance, a list of places she had been that week). And suppose $\mathcal{R}$ has a method $x()$ that can state one valid value for that statement (e.g.

where she was last night). Can the government compel $\mathcal{R}$ to reveal where she was sometime this past week? Or, in our parlance, is the target action which returns $\mathcal{R}.x()$ entailable? (For the sake of this example, suppose the government was permitted to compel *any* act they could verify, even explicit verbal testimony.)

If the government has no evidence that the respondent was located anywhere in particular in the last week, then intuitively, there should be no way to entail the action of $\mathcal{R}.x()$.[20] Theorem 5.4 makes this idea precise.

EVIDENCE $\mathcal{E}_{\text{lang}}$:
Variables: $\mathcal{R}.z$
Method: $\mathcal{R}.x()$
assert $\mathcal{R}.x() \in L_{\mathcal{R}.z}$ **where**
  each $z' \in \{0,1\}^*$ is associated with a language $L_{z'} \subseteq \{0,1\}^*$

TARGET $\mathcal{T}^{\mathcal{R}}_{\text{lang}}$
return $\mathcal{R}.x()$

Theorem 5.4. For a given $\mathcal{N}$ and $\mathcal{E}$, let $\mathfrak{R}_{\mathcal{N},\mathcal{E}} = \{\mathcal{R} : (\mathcal{R}, \mathcal{N}) \in \mathcal{E}\}$ be the set of respondents that are consistent with $\mathcal{N}$ and $\mathcal{E}$. For $\mathcal{E}' \geq \mathcal{E}_{\text{lang}}$, if $\exists \mathcal{N}$ such that $\mathfrak{R}_{\mathcal{N},\mathcal{E}'} \neq \emptyset$ and

$$\bigcap_{\mathcal{R} \in \mathfrak{R}_{\mathcal{N},\mathcal{E}'}} L_{R.z} = \emptyset,$$

then there does not exist a demonstrable $\mathcal{V}$ which entails $\mathcal{T}_{\text{lang}}$ with respect to $\mathcal{E}'$.

Note that the hypothesis of Theorem 5.4 implies that the government has no ability to check membership in the language $L_{\mathcal{R}.z}$. This is the formalization of the property stated above, that the government does not know "what it is looking for."

We defer the proof of Thm. 5.4 to Appendix B.1. The proof does not rely on any computational limitations on the part of the government—only the information-theoretic uncertainty of the true value of $\mathcal{R}.z$.

5.2 Impossibility of entailing distributions

In Appendix A.3 we will investigate actions that result in a distribution, such as compelled encryption or commitments. In those sections, it will be useful to make use of the following result, which states that it is impossible to entail actions that result in a distribution using the random coins of the action itself.

Theorem 5.5. Let $\mathcal{E}$ be some evidence, and let $\mathcal{T}_{\text{rand}}$ be some target action. Suppose that $\mathcal{T}_{\text{rand}}$ uses its own randomness in a non-trivial way: specifically, there exists $\mathcal{E}$-consistent $\mathcal{N}, \mathcal{R}$ with the property that at least one fixed setting of the random tapes of $\mathcal{N}, \mathcal{R}$ have the property that $|\text{Support}(\mathcal{T}^{\mathcal{N},\mathcal{R}}_{\text{rand}})| \geq 2$. Then, there is no demonstrable $\mathcal{V}$ that entails $\mathcal{T}_{\text{rand}}$.

We defer the proof of Theorem 5.5 to Appendix B.2.

[20] Observe that there are demonstrable $\mathcal{V}$s with "return $\mathcal{R}.x()$" as the exemplar. However, the respondent may respond by, for example, responding to the query with "Boston" even if she had never been there. This is not the result of $\mathcal{R}.x()$, but the verifier must accept it anyway because the government has no way to rule out the possibility.

5.3 Entailment and partial specifications

The ability for evidence to partially-specify functionalities in $\mathcal{N}$ poses a major difficulty for proving entailment in many cases.

The algorithms in Figure 1 illustrate the problem. Consider $\mathcal{E}_{\text{read}}$ stating that $D_{\text{read}} \preccurlyeq \mathcal{N}[\text{DLoc}]$: there is a device at location DLoc in nature that is consistent with D_{read}. D_{read} has a variable m and a method READ() that returns m. It may seem that $\mathcal{T}_{\text{read}}$ which simply outputs $\mathcal{N}[\text{DLoc}].\text{READ}()$ should be entailable. After all, $\mathcal{P}$ could just call $\mathcal{N}[\text{DLoc}].\text{READ}()$ itself. Whatever $\mathcal{V}$ entails $\mathcal{T}_{\text{read}}$ could have an exemplar action that does nothing.

But in fact $\mathcal{T}_{\text{read}}$ is not entailable! The problem is that the behavior of $\mathcal{N}[\text{DLoc}]$ is only partially specified. While it must be consistent with D_{read}, it may have other methods too. For example, $\mathcal{N}[\text{DLoc}]$ could instead contain device $D_{\text{readWrite}}$ (Figure 1) which has another method WRITE(x) that sets m to x. Because $\mathcal{E}$ contains no information about m, if the $\mathcal{A}^*$ that does nothing is $\mathcal{V}$-conforming, then $\mathcal{A}_x$ that calls WRITE(x) is also $\mathcal{V}$-conforming.

As such, many of our entailment claims will be proved with respect to the stronger evidence $\mathcal{E}_D$ wherein some function specifications are strengthened from partial to full specifications (see Definition 4.4). This showcases our framework's ability to reason about uncertainty about the world, as illustrated by the example of deniable encryption in Section 6.2.

6 COMPELLED ACTS OF COMPUTATION

In this section we consider how to construct demonstrable verifiers and analyze entailment for some important idealized foregone conclusion scenarios. We begin with the common example of compelling a respondent to enter a passcode that decrypts a device under the assumption that the encryption scheme is not deniable (Section 6.1). Then in Section 6.2 we show that removing the assumption of non-deniability eliminates the ability to entail the action of "honest" decryption, standing in contrast with most prior legal analysis. We then describe the entailability of cryptographic *decommitment*, which implies entailment of a wide variety of functionalities (Section 6.3). This finding stands in direct contrast with the approach of Scheffler and Varia [32] (see Appendix 7.2). We also examine additional examples, namely two-factor authentication, hash preimages, *en*cryption, and commitmnets in Appendix A.

6.1 Entering a password to decrypt

In the archetypal compelled decryption case, the government has lawfully seized an encrypted device and seeks order compelling the respondent to decrypt by entering her password. The government wants to specify an action for which all implicit testimony is foregone. For this act to be meaningful, it should allow the government to recover the decrypted plaintext. In our framework, the government seeks a demonstrable $\mathcal{V}_{\text{pwd}}$ that entails the target task $\mathcal{T}_{\text{pwd}}$ of decrypting and producing the plaintext (given in Figure 2). For an idealized non-deniable encryption, we will show that this is possible if the government knows that the respondent knows the password (as in Kerr [24]).

In this subsection, we will consider the algorithms and evidence given in Figure 2 (unless otherwise stated). In particular, we consider a simple password-protected device D_{pwd} that provides read access to a message if the correct password is entered, and hides

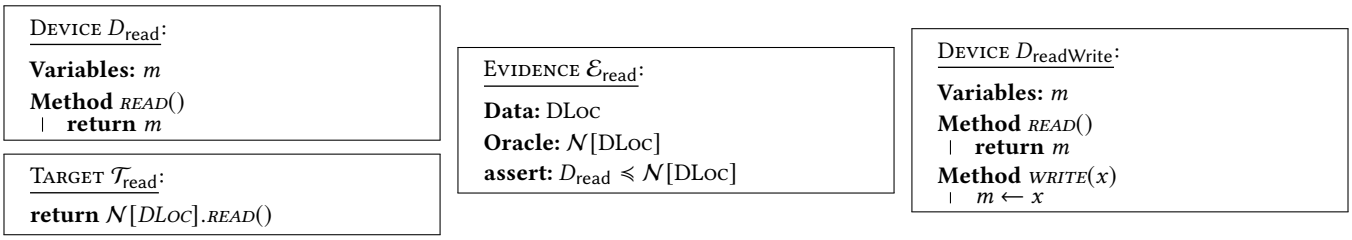

Figure 1: Algorithms for the example in Section 5.3 to show the challenge of entailment with partial evidence

all message contents otherwise. If the government has evidence $\mathcal{E}_{\text{pwd}}$ that the respondent knows this password, then it can use a verifier $\mathcal{V}^{\mathcal{N}}_{\text{pwd}}$ that checks if the device is unlocked and capable of displaying any message, along with an exemplar action $\mathcal{A}^{*\mathcal{N},\mathcal{R}}_{\text{pwd}}$ in which the respondent types the known password into the prompt.

6.1.1 $\mathcal{R}$ knows the password. We begin with the case where $\mathcal{G}$ knows that $\mathcal{R}$ knows the password. The evidence $\mathcal{E}_{\text{pwd}}$ states that $\mathcal{G}$ knows that a device implementing D_{pwd} is at location DLoc in nature. $D = D_{\text{pwd}}$ is some idealized encrypted device, with a hard-coded password $D.pwd$ and message $D.m \neq \bot$. The evidence also states that $\mathcal{R}$ has some method $\mathcal{R}.pwd()$ that returns the password $D.pwd$ (the starred line in Fig. 2). After that password is entered into D, the message $D.m$ can be read using the method D.READ().

Applying our framework, the government must specify a demonstrable verification procedure $\mathcal{V}_{\text{pwd}}$. The government's verifier $\mathcal{V}_{\text{pwd}}$ reads m from the device, outputting ACCEPT if $m \neq \bot$.

Claim 6.1. $\mathcal{V}_{\text{pwd}}$ is demonstrable with respect to $\mathcal{E}_{\text{pwd}}$.

PROOF. Consider exemplar action $\mathcal{A}^*_{\text{pwd}}$ in Fig. 2. First, we must check that every method call by $\mathcal{A}^*_{\text{pwd}}$ to $\mathcal{R}$ produces some output. $\mathcal{A}^*_{\text{pwd}}$ only calls $\mathcal{R}.pwd()$. By $\mathcal{E}$, this method exists and outputs $D.pwd$. Second, we must check that $\mathcal{V}_{\text{pwd}}$ accepts $\mathcal{A}^*_{\text{pwd}}$ with respect to $\mathcal{R}$, $\mathcal{N}$. Let M be the device at $\mathcal{N}$[DLoc]. $\mathcal{E}$ guarantees that M implements D_{pwd} (i.e., $D_{\text{pwd}} \preccurlyeq M$). $\mathcal{A}^*$ calls M.PROMPT with input $\mathcal{R}.pwd()$. By $\mathcal{E}$, $\mathcal{R}.pwd() = M.pwd$. Hence, when $\mathcal{A}^*$ halts, $M.decrypted$ = TRUE. Now $\mathcal{V}_{\text{pwd}}$ calls M.READ(), which returns $M.m$. $\mathcal{E}$ states that $m \neq \bot$. Hence $\mathcal{V}_{\text{pwd}}$ returns ACCEPT. □

By construction, any $\mathcal{V}_{\text{pwd}}$-conforming action $\mathcal{A}$ will allow the government to recover the plaintext m, assuming the encrypted device is not deniable. We use entailment to make this precise. Consider $\mathcal{T}_{\text{pwd}}$ in Figure 2, which enters $\mathcal{R}.pwd()$ into the device's password prompt, then reads and returns the message m. By the evidence, the result is the plaintext message.

We prove that $\mathcal{V}_{\text{pwd}}$ entails $\mathcal{T}_{\text{pwd}}$ under the stronger evidence $\mathcal{E}_{D_{\text{pwd}}}$ that the device is fully specified by D—ruling out deniability. This restriction is necessary, and follows the general pattern described in Section 5.3. For one, the proof breaks down: the step marked (♣) does not hold for arbitrary $\mathcal{E}$. Moreover, Section 6.2 shows that the government cannot entail $\mathcal{T}_{\text{pwd}}$ when the device is deniable but otherwise consistent with D_{pwd}.

This example illustrates the expressiveness of our approach and the power of entailment. Claim 6.2 states that in this example 'decrypting by entering a password' entails 'decrypting then producing the plaintext'. But the act of decrypting-then-producing is not itself compellable! Specifically, it would require implicit testimony about conformity that is not a foregone conclusion under the evidence. (Additionally outside of our model, some would argue that producing the plaintext presents testimonial issues beyond *Fisher*.)

Claim 6.2. $\mathcal{V}_{\text{pwd}}$ entails $\mathcal{T}_{\text{pwd}}$ with respect to $\mathcal{E}_{D_{\text{pwd}}}$.

PROOF. We must provide an efficient oracle machine $\mathcal{P}_{\text{pwd}}$ such that for all $\mathcal{A}$ that is $\mathcal{V}_{\text{pwd}}$-conforming: $\mathcal{P}^{\mathcal{N}'}_{\text{pwd}}(\tau) = \mathcal{T}^{\mathcal{N},\mathcal{R}}_{\text{pwd}}$. Let $M = \mathcal{N}$[DLoc] and $M' = \mathcal{N}'$[DLoc]. Let $\mathcal{P}^{\mathcal{N}'}_{\text{pwd}}$ simply call and return M'.READ() as in Figure 2. To complete the proof, we show that both $\mathcal{P}_{\text{pwd}}$ and $\mathcal{T}_{\text{pwd}}$ always output $M.m$.

(RHS). By $\mathcal{E}_{\text{pwd}}$-consistency of $\mathcal{R}$, we get $x = M.pwd$ on line 1 of $\mathcal{T}_{\text{pwd}}$. By $\mathcal{E}$-consistency of M, we get $m' = M.m$ on line 3 of $\mathcal{T}_{\text{pwd}}$. Hence $\mathcal{T}^{\mathcal{N},\mathcal{R}}_{\text{pwd}}$ always outputs $M.m$.

(LHS). Because $\mathcal{A}$ is $\mathcal{V}_{\text{pwd}}$-conforming, we have $\langle \mathcal{V}^{\mathcal{N}}_{\text{pwd}}, \mathcal{A}^{\mathcal{N},\mathcal{R}} \rangle$ = ACCEPT. By construction, the execution $\langle \cdot, \cdot \rangle$ first runs $\mathcal{A}$ and then runs $\mathcal{V}_{\text{pwd}}$. Because $\mathcal{V}_{\text{pwd}}$ does not change the state of M (i.e., does not call M.PROMPT), $\mathcal{N}'$ is equal to the state of nature after $\mathcal{A}^{\mathcal{N},\mathcal{R}}$ terminates. By construction of $\mathcal{V}_{\text{pwd}}$ (and the conformity of $\mathcal{A}$), M'.READ() $\neq \bot$ after $\mathcal{A}$ terminates. If M'.READ() $\neq \bot$, then M'.READ() $= M.m$ (♣). Hence $\mathcal{P}^{\mathcal{N}'}_{\text{pwd}}$ always outputs $M.m$. □

6.1.2 $\mathcal{R}$ may not know the password. Next, we drop the starred assertion in $\mathcal{E}_{\text{pwd}}$ in Figure 2. This tweaks the facts to remove the government $\mathcal{G}$'s assertion that $\mathcal{R}$ knows a password to D. Call the modified evidence $\mathcal{E}^\star$.

$\mathcal{E}^\star$ states that the respondent has some method $\mathcal{R}.pwd()$ that may do something. But not what it does, nor whether it has any relation to D. Under $\mathcal{E}^\star$, it is entirely consistent that $\mathcal{R}$ simply does nothing. That is, for any $(\mathcal{N}, \mathcal{R})$ that is $\mathcal{E}^\star$-conforming, $(\mathcal{N}, \mathcal{R}_\bot)$ is also $\mathcal{E}^\star$-conforming where $\mathcal{R}_\bot$ immediately halts on all inputs.

Claim 6.3. $\mathcal{T}_{\text{pwd}}$ is not entailable with respect to $\mathcal{E}^\star$.

PROOF. Let $(\mathcal{N}, \mathcal{R}_{\text{pwd}})$ be $\mathcal{E}_{\text{pwd}}$-consistent. Because $\mathcal{E}_{\text{pwd}} \geq \mathcal{E}^\star$, $(\mathcal{N}, \mathcal{R}_{\text{pwd}})$ is $\mathcal{E}^\star$-consistent. Hence $(\mathcal{N}, \mathcal{R}_\bot)$ is also $\mathcal{E}^\star$-consistent, where $\mathcal{R}_\bot$ immediately halts on all inputs. Fix $\mathcal{V}$ demonstrable and $\mathcal{A}$ that is $\mathcal{V}$-conforming, all with respect to $\mathcal{E}^\star$. Because $\mathcal{R}_\bot$ is $\mathcal{E}^\star$-consistent, the action $\mathcal{A}_\bot$ which emulates $\mathcal{A}$ with $\mathcal{R}_\bot$ is also $\mathcal{V}$-conforming with respect to $\mathcal{E}^\star$.

For any post-processor $\mathcal{P}$, the output of $\mathcal{P}$ after $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{\mathcal{N},\mathcal{R}}_\bot \rangle$ is independent of $\mathcal{R}$. On the other hand, $\mathcal{T}^{\mathcal{N},\mathcal{R}_{\text{pwd}}}_{\text{pwd}}$ always outputs $D.m \neq \bot$, while $\mathcal{T}^{\mathcal{N},\mathcal{R}_\bot}_{\text{pwd}}$ always outputs $\bot$. Hence, there exists $\mathcal{E}^\star$-consistent $(\mathcal{N}, \mathcal{R})$ such that $\mathcal{P}$ does not always output the result of $\mathcal{T}^{\mathcal{N},\mathcal{R}}_{\text{pwd}}$ (e.g., $\mathcal{R}$ is an appropriately chosen distribution over $\mathcal{R}_\bot$ and $\mathcal{R}_{\text{pwd}}$). Hence, $\mathcal{T}_{\text{pwd}}$ is not entailable with respect to $\mathcal{E}^\star$. □

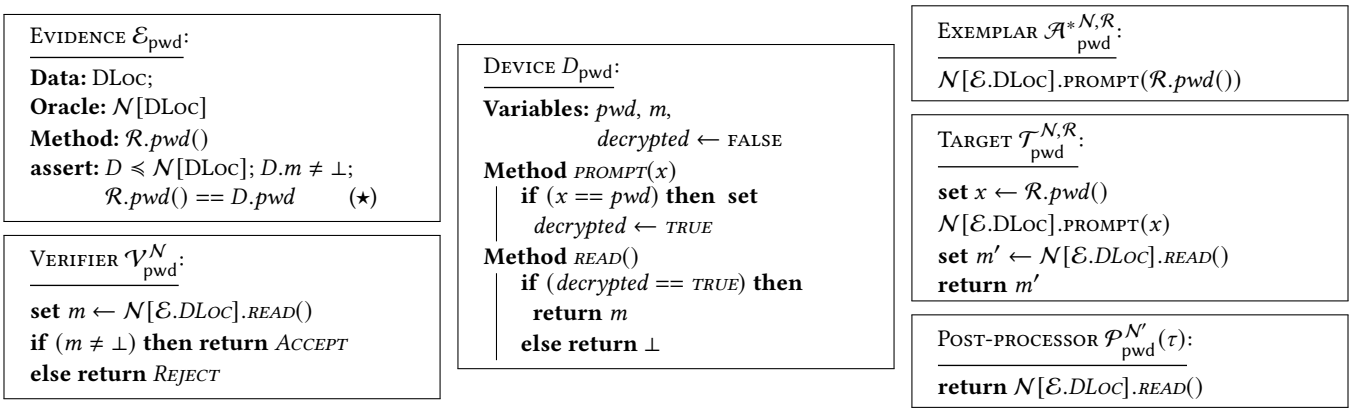

Figure 2: Example algorithms for compelled decryption by entering a password as described in Section 6.1 and used in Section 6.2. The starred ($\star$) assertion in $\mathcal{E}_{\text{pwd}}$ refers to the respondent's knowledge of the decryption password. Removing that line captures the setting when the government has no evidence of the respondent's knowledge of the password.

6.2 Deniable encryption

We continue the example from Section 6.1 and Figure 2 wherein the government wishes to compel decryption by entering a password. Now we consider the case of deniable encryption. Deniable encryption introduces a *duress* password which, when entered, allows a device to be decrypted to something other than its "true" contents. For illustrative purposes, we analyze a simple but unrealistically powerful form of deniable encryption D_{deny}, wherein the duress password can be used to overwrite the device's contents arbitrarily.

Device D_{deny} implements D_{pwd}:

```
// Has the following variables and methods in
   addition to those provided by D_pwd in Fig. 2
```

Variables: *duress*
Method DENY(x, y)
  if $(x ==$ *duress*$)$ **then set** $m \leftarrow y$
  set *decrypted* $\leftarrow$ TRUE

In Claim 6.2, we proved entailment only with respect to the strong evidence $\mathcal{E}_{D_{\text{pwd}}}$ that states that the encryption is *not* deniable. This is not a fluke. Consider the idealized version of deniable encryption in D_{deny}. Because $D_{\text{pwd}} \preccurlyeq D_{\text{deny}}$, it is entirely consistent with $\mathcal{E}_{\text{pwd}}$. So the possibility that the actual encrypted device uses a form of deniable encryption affects entailment. Absent stronger evidence that the device is not deniable (i.e.: $\mathcal{E}_{D_{\text{pwd}}} \geq \mathcal{E}_{\text{pwd}}$), $\mathcal{T}_{\text{pwd}}$ is not entailable, and the message that will ultimately be read from D is not meaningful.

While entailment fails, demonstrability does not. The same $\mathcal{V}_{\text{pwd}}$ is demonstrable with the same exemplar action $\mathcal{A}^*{}_{\text{pwd}}$. Using our framework, a respondent may ultimately be compelled to enter a password *even if the government knows the encryption is deniable*—but the respondent would be free to use the duress password. This is in contrast to prior legal approaches to this fact pattern, as discussed at the end of Section 1.

6.3 Opening a commitment

Next, we consider *commitments*—cryptographic objects related to, but distinct from encryption. In this section, we ask: when can the government compel the opening of a commitment? In our framework, the answer is: when the government has the commitment and knows that the respondent can open it. Note that this answer generalizes the claims about committing encryption from Section 6.1.

At a high level, commitments are like lockboxes, where each box has a bespoke lock and key. Once you put a secret message into the lockbox, nobody can read the secret (hiding), and nobody can change the secret (binding). Whoever has the key can open the box in front of others, thereby convincing them that box always contained that particular secret message. An important difference between commitments and encryption is this public-opening property. For instance, deniable encryption schemes are specifically non-committing: it can be decrypted in different ways. Commitments can only be opened a single way.

Moving from the physical analogy to the computational setting, commitments schemes provide an algorithm to *commit* to a secret x, producing the *commitment* c and the *decommitment* d. *Opening* the commitment means revealing d and x. Given c, d, and x, anybody can *check* that the opening was done correctly.

Definition 6.4 (String commitment scheme)**.** Let $(\text{Com}, \text{Check})$ be a string commitment scheme where $(c, d) \leftarrow \text{Com}(x; r)$ yields a pair of strings (c, d) called the *commitment* and *decommitment* to message x respectively, and $\text{Check}(c', d', x')$ returns either 1 or 0. For all r, if $(c, d) \leftarrow \text{Com}(x; r)$ then $\text{Check}(c, d, x) = 1$.

Binding (informal): A commitment scheme is *binding* if it is difficult to find (x, x', d, d', c) such that $x \neq x$, $\text{Check}(c, d, x) = 1$, and $\text{Check}(c, d', x') = 1$. A commitment scheme is *perfectly binding* if there does not exist such a tuple (x, x', d, d', c).

Hiding (informal): Suppose $(c, d) \leftarrow \text{Com}(x; r)$. A commitment scheme is *hiding* if it is difficult to find x given only c. A commitment scheme is *perfectly hiding* if the distribution of c is identical for any x, x' input into Com (over the choice of r).

Consider the Evidence $\mathcal{E}_{\text{decom}}$, Verifier $\mathcal{V}_{\text{decom}}$, Target Action $\mathcal{T}_{\text{decom}}$, and Exemplar Action $\mathcal{A}^*{}_{\text{decom}}$ given in Figure 3. $\mathcal{E}_{\text{decom}}$ states that that the respondent is able to produce a valid opening to the commitment c at location $\mathcal{N}[\text{COMMLOC}]$. The verifier $\mathcal{V}_{\text{decom}}$ receives (x', d') from the respondent (by way of a conforming

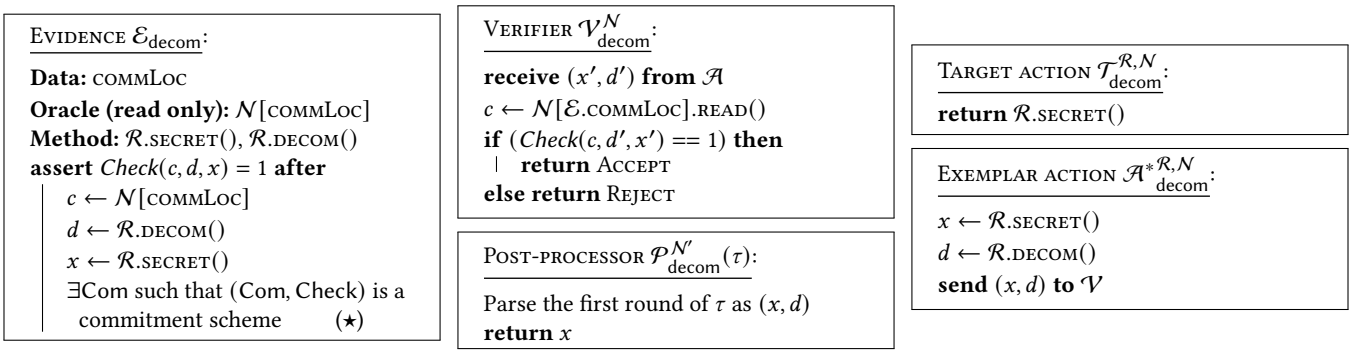

Figure 3: Evidence, Verifier, Target Action, and Exemplar Action for decommitment (Section 6.3). We define $\mathcal{E}_{\text{bind}} \geq \mathcal{E}_{\text{decom}}$ to be the stronger evidence where the commitment scheme on the starred (⋆) line is *perfectly binding*.

action $\mathcal{A}$), fetches c from nature, and runs the commitment's Check algorithm. The exemplar $\mathcal{A}^*$ outputs (x, d) produced by $\mathcal{R}$.

Claim 6.5. $\mathcal{V}_{\text{decom}}$ is demonstrable with respect to $\mathcal{E}_{\text{decom}}$ with exemplar action $\mathcal{A}^*{}_{\text{decom}}$.

Proof. If $\mathcal{N}, \mathcal{R}$ are $\mathcal{E}_{\text{decom}}$-consistent, then $\mathcal{A}^*{}_{\text{decom}}$ will result in $\mathcal{V}$ accepting, because the Evidence states that Check(C.read(), $\mathcal{R}.decom()$, $\mathcal{R}.secret()$) = 1. We know $\mathcal{R}$ has the ability to perform this action, because it involves only a read to $\mathcal{R}.secret()$ and $\mathcal{R}.decom()$ which were specified in $\mathcal{E}_{\text{decom}}$. Thus, $\mathcal{V}_{\text{decom}}$ is demonstrable with witness action $\mathcal{A}^*{}_{\text{decom}}$. □

In fact, this holds even without binding. That is, dropping the starred (⋆) line in $\mathcal{E}_{\text{decom}}$ in Figure 3 does not affect demonstrability.

If the government additionally has evidence that the commitment is perfectly binding, then $\mathcal{V}_{\text{decom}}$ entails $\mathcal{T}_{\text{decom}}$ which discloses $\mathcal{R}$.secret().[21] Let $\mathcal{E}_{\text{bind}} \geq \mathcal{E}_{\text{decom}}$ represent this stronger evidence. (Both $\mathcal{T}_{\text{decom}}$ and $\mathcal{E}_{\text{bind}}$ are defined in Fig. 3.

Claim 6.6. If Com is perfectly binding, then $\mathcal{V}_{\text{decom}}$ entails $\mathcal{T}_{\text{decom}}$ under $\mathcal{E}_{\text{bind}}$.

Proof. Recall that $\mathcal{E}_{\text{decom}}$ reads c from $\mathcal{N}[commLoc]$. $\mathcal{E}_{\text{decom}}$ asserts that for $d \leftarrow \mathcal{R}$.decom() and $x \leftarrow \mathcal{R}$.secret(), it must be true that Check$(c, d, x) = 1$. Recall that $\mathcal{V}_{\text{decom}}$ reads $c' \leftarrow \mathcal{N}[\mathcal{E}_{\text{decom}}.commLoc]$, and since this oracle is read-only, we have that $c' = c$. Since the commitment scheme is perfectly binding, there is no (x, x', d, d', c) such that $x \neq x'$, and both Check$(c, d, x) = 1$ and Check$(c, d', x') = 1$. Thus, $x' = x$ and $d' = d$.

Letting $\mathcal{N}'$ be the state of Nature after running $\mathcal{A}$, the post-processor $\mathcal{P}^{\mathcal{R},\mathcal{N}'}_{\text{decom}}$ which returns the x' sent by $\mathcal{A}$ will always return $\mathcal{R}.secret()$, thus entailing $\mathcal{T}_{\text{decom}}$. □

A corollary of Claim 6.6 is that, under the same evidence, the government can entail arbitrary computation on the committed secret $\mathcal{R}$.secret(). For any f, $\mathcal{T}_f$ is entailable with respect to $\mathcal{E}_{\text{decom}}$:

$$\mathcal{T}_f^{\mathcal{N},\mathcal{R}} := \textbf{return } f(\mathcal{R}.\text{secret}())$$

[21] If the commitment scheme is only computationally binding, then we expect entailment would still hold against a computationally-bounded respondent along the lines of Claim A.3.

Remark 6.7. In this section, we showed that compelling the opening of a unknown but committed secret is demonstrable and entailable if the scheme is perfectly binding. In Appendix A.3, we show that there is no way to entail a *commitment* of an unknown secret. This is reassuring. If it were not true, then $\mathcal{G}$ could entail commitment-then-open of arbitrary unknown secrets. It is worth noting that there are many demonstrable $\mathcal{V}$s that have the *exemplar action* of committing to a secret value. However, the fact that none of those $\mathcal{V}$s *entail* the action of committing to a secret value means that those $\mathcal{V}$s would also be satisfied by other actions, possibly for example committing to the all-zeros string.

7 COMPARISON TO PRIOR WORK

7.1 Brief comparison to other legal scholarship

Here, we compare our verification-centric view to the body of legal scholarship on compelled decryption. On the one hand, scholars like Kiok [26] and Sacharoff [31] posit that the government can compel decryption only if they know the desired files with "reasonable particularity," and Winkler [51] argues that the self-incrimination privilege provides an absolute defense against compelled decryption. On the other hand, Cohen and Park [6], Kerr [23–25], McGregor [27], and Terzian [37, 38] deem compelled decryption to be a foregone conclusion in some contexts, either to re-balance government power due to the widespread use of encryption or because all the testimony implicit in the act of decryption (e.g., ownership, knowledge of password) is foregone.

Our framework largely agrees with the latter interpretation of the foregone conclusion, by considering *all acts* involved in the performance of decryption rather than simply the files produced. That said, our framework reaches a novel conclusion about the compellability of "deniable encryption" schemes, which include special duress password(s) that can reveal different files. Here, Kerr and Cohen-Park reach opposite conclusions. By separating demonstrability from entailment, our framework reaches a "middle ground" where decryption can be compelled, but the respondent can use any password unless the files are known with reasonable particularity.

7.2 Comparison to Scheffler and Varia

The work that is most related to ours is the paper of Scheffler and Varia [32], which also contributes a cryptography-inspired candidate definition of the foregone conclusion doctrine. This work

and [32] agree on landmark Supreme Court and circuit court decisions to date about the foregone conclusion doctrine [12, 13, 16–18, 45, 46, 49]. As a general rule of thumb that we expect to hold for many realistic fact patterns, an action that is a foregone conclusion under [32] is likely entailable in our current framework. A notable exception is the inability for our current framework to entail random sampling from a probability distribution (Theorem 5.5).

That being said, the two works differ in several legal and computer science aspects.

First, this work requires the government to specify how it will *verify* the respondent's act, thus ensuring that the government is not "relying on the truthtelling" [13, l. 411]. In contrast, [32] requires the government to specify how it *could have performed* the respondent's act using the cryptographic idea of simulation but without access to the "contents of [the respondent's] mind" [13, l. 421], thus ensuring that the respondent's production "adds little or nothing to the sum total of the Government's information" [13, l. 411].

Second, the two works take different approaches to conformity. Scheffler and Varia dealt with the implicit testimony from conformity using simulation. In contrast, this work defines away the problem of conformity by allowing the respondent to perform *any* conforming action.

Third, the modeling in this work allows modeling more scenarios than [32]. Scheffler and Varia focused on acts of *production*, and involved the simulation of *what was produced*. Our work, on the other hand, focuses on acts of either production or performance, and can analyze actions taken that have impact on the world even if no item is produced to the court as a result. To accomplish this technically, rather than modeling the world as a collection of passive strings such as a ciphertext stored on a smartphone's hard drive, we model dynamic, stateful, interactive processes such as the phone itself and all of the ways that one can interact with it.

The archetypal compelled decryption scenario asks: When can a respondent be compelled to enter a password into an encrypted device in the government's possession? Scheffler and Varia model a scenario where the respondent performs password-based decryption and produces the plaintext. But Cohen and Park argue that the "decryption" scenario is doctrinally distinct from the "production of plaintext" [6]. In contrast to [32], our current model allows us to consider either scenario directly.

8 CONCLUSION & DISCUSSION

In summary, we adopt a view of implicit testimony that treats ability and conformity as equally important, and we present a new verification-centric approach to compelling acts while constructively satisfying the foregone conclusion doctrine. On the technical side, we formally define the concepts of demonstrability and entailment and explore the compellability of acts of cryptography within our framework. Our goal in this project was CS-*and*-law: rigorous computer science that substantively engages with and contributes to legal thought. In this section, we take a step back and reflect on what seemed to work for us.

This paper is the culmination of a five-year research agenda that alternated between primarily legal and primarily technical contributions. First, AC in collaboration with Sunoo Park produced a law-first analysis grounded in their knowledge of cryptography; they showed how the application of the foregone conclusion doctrine can depend on the nuances of the underlying technology [6]. Next, SS and MV produced a technical-first analysis inspired by the law; they gave a more restricted simulation-based formal framework and analyzed how the compellability of different cryptosystems under it [32]. All three authors then worked to reconcile the differing accounts of those two works, resulting in a new legal analysis that introduced the idea that ability and conformity are testimony implicit in acts of performance [7]. The present paper represents our effort to turn these ideas into a formal framework; more than just a formalism, that effort spawned our verification-centric approach and the concepts of demonstrability and entailability.

In this project, neither side was subservient to the other. We did not start with a legal view or a set of technical outcomes to which we molded our framework.

Instead, our meta-level approach in this project was an iterative, alternating application of legal and technical thinking. At a high level, our goal was to fill in a gap in our understanding of the foregone conclusion doctrine: namely, it's requirement that the government may not "rely on the truthtelling" of the respondent. First, we would state a non-mathematical version of the requirement based on our understanding of the case law and fact patterns considered so far. Second, we attempted to formalize that statement in a technically precise and sound way, building from the framework of [32]. Third, we would test the result against many fact patterns, whether uncontroversial act-of-production cases, existing compelled decryption cases, the cryptographic primitives studied in this paper, or hypotheticals specifically designed to stress the framework. Fourth, we would ask whether the formalism uncovered a new aspect of the legal question that could help refine our thinking. After many many iterations, we converged on the ideas in this paper.

We do not know what legal questions are amenable to the sort of CS-and-law analysis we undertake in this paper. We highlight some properties that we believe make compelled decryption fruitful for a cross-disciplinary study, in the hopes that it may help others find other fertile grounds. First, law alone offers no obvious answers. Scholars are in disagreement and state supreme courts are split. And as Cohen and Park argue, forgone conclusion analyses can depend on technical details, and reasoning by analogy can only go so far. Second, there are clear connections to well-studied concepts in cryptography and computer science. Of course the very artifacts in question issue are technological and cryptographic. And at a conceptual level, cryptographic community has spent decades stress-testing notions of simulation and verification. The intuitive connection to ideas in *Fisher*'s — adding to the government's knowledge, or relying on the respondent's truthtelling — are immediate to a cryptographer (though making the intuition precise is not easy). Third, the results of this project is of value to both communities, with the potential to affect future court cases and to spur new research directions in security and cryptography.

ACKNOWLEDGMENTS

Aloni Cohen and Mayank Varia were supported by the National Science Foundation under Grant No. 1915763 and by the DARPA

SIEVE program under Agreement No. HR00112020021. Mayank Varia was additionally supported by National Science Foundation Grants No. 1718135, 1801564, and 1931714. Sarah Scheffler was supported by a Google Ph.D. Fellowship and the Center for Information Technology Policy at Princeton University. We are grateful for the feedback from participants at the 2021 Privacy Law Scholar's Conference and 2022 CTIC Law & Computer Science Roundtable.

REFERENCES

[1] Micah Altman, Aloni Cohen, Kobbi Nissim, and Alexandra Wood. 2021. What a hybrid legal-technical analysis teaches us about privacy regulation: The case of singling out. *BU J Sci. & Tech. L.* 27 (2021), 1.

[2] Ran Canetti. 2001. Universally Composable Security: A New Paradigm for Cryptographic Protocols. In *42nd FOCS*. IEEE Computer Society Press, 136–145. https://doi.org/10.1109/SFCS.2001.959888

[3] Ran Canetti, Asaf Cohen, and Yehuda Lindell. 2015. A Simpler Variant of Universally Composable Security for Standard Multiparty Computation. In *CRYPTO 2015, Part II (LNCS, Vol. 9216)*, Rosario Gennaro and Matthew J. B. Robshaw (Eds.). Springer, Heidelberg, 3–22. https://doi.org/10.1007/978-3-662-48000-7_1

[4] Aloni Cohen, Moon Duchin, J. N. Matthews, and Bhushan Suwal. 2021. Census TopDown: The Impacts of Differential Privacy on Redistricting. In *FORC (LIPIcs, Vol. 192)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 5:1–5:22.

[5] Aloni Cohen and Kobbi Nissim. 2020. Towards formalizing the GDPR's notion of singling out. *Proc. Natl. Acad. Sci. USA* 117, 15 (2020), 8344–8352.

[6] Aloni Cohen and Sunoo Park. 2018. Compelled Decryption and the Fifth Amendment: Exploring the Technical Boundaries. *Harvard Journal of Law & Technology* 32 (2018), 169–234. Issue 1.

[7] Aloni Cohen, Sarah Scheffler, and Mayank Varia. 2021. Telling the Truth about Compelled Encryption and Contents of the Mind *(Privacy Law Scholars Conference)*.

[8] Commonwealth v. Davis, Pa: Supreme Court, Middle Dist. 2019.

[9] Commonwealth v. Gelfgatt, 11 N.E.3d 605 (Mass.). 2014.

[10] Commonwealth v. Jones, 481 Mass. 540 - Mass: Supreme Judicial Court. 2019.

[11] Curcio v. United States, 354 U.S. 118. 1957.

[12] Doe v. United States, 487 U.S. 201. 1988.

[13] Fisher v. United States, 425 U.S. 391. 1976.

[14] Sanjam Garg, Shafi Goldwasser, and Prashant Nalini Vasudevan. 2020. Formalizing Data Deletion in the Context of the Right to Be Forgotten. In *EUROCRYPT 2020, Part II (LNCS, Vol. 12106)*, Anne Canteaut and Yuval Ishai (Eds.). Springer, Heidelberg, 373–402. https://doi.org/10.1007/978-3-030-45724-2_13

[15] Holt v. United States, 218 U.S. 245. 1910.

[16] In re Grand Jury Proceedings, 41 F. 3d 377 (8th Cir.). 1994.

[17] In re Grand Jury Subpoena, 383 F.3d 905 (9th Cir.). 2004.

[18] In re Grand Jury Subpoena Duces Tecum, 1 F. 3d 87 (2nd Cir.). 1993.

[19] In re Grand Jury Subpoena Duces Tecum Dated March 25, 2011 (United States v. Doe), 670 F.3d 1335 (11th Cir.). 2012.

[20] In re Grand Jury Subpoena to Sebastetien Boucher, No. 2:06-mJ-91, 2009 WL 424718. 2009.

[21] In re Search of a Residence in Aptos, California 95003, No.17-mj-70656-JSC-1, 2018 WL 1400401. 2018.

[22] Samuel Judson and Joan Feigenbaum. 2022. On Heuristic Models, Assumptions, and Parameters. *CoRR* abs/2201.07413 (2022).

[23] Orin Kerr. 2016. Opinion: The Fifth Amendment limits on forced decryption and applying the 'foregone conclusion' doctrine. https://www.washingtonpost.com/news/volokh-conspiracy/wp/2016/06/07/the-fifth-amendment-limits-on-forced-decryption-and-applying-the-foregone-conclusion-doctrine/.

[24] Orin S Kerr. 2018. Compelled Decryption and the Privilege Against Self-Incrimination. *Tex. L. Rev.* 97 (2018), 767.

[25] Orin S Kerr. 2020. Decryption Originalism: The Lessons of Burr. *Available at SSRN* (2020).

[26] Jeffrey Kiok. 2015. Missing the Metaphor: Compulsory Decryption and the Fifth Amendment. *Boston University Public Interest Law Journal* 24 (2015), 53–80. Issue 1.

[27] Nathan K. McGregor. 2010. The Weak Protection of Strong Encryption: Passwords, Privacy, and Fifth Amendment Privilege. *Vanderbilt Journal of Entertainment & Technology Law* 12 (2010), 581–609. Issue 3.

[28] Robbie Morrison, Natasha CHL Mazey, and Stephen C Wingreen. 2020. The DAO controversy: the case for a new species of corporate governance? *Frontiers in Blockchain* 3 (2020), 25.

[29] Kobbi Nissim. 2021. Privacy: From Database Reconstruction to Legal Theorems. In *PODS*. ACM, 33–41.

[30] Kobbi Nissim, Aaron Bembenek, Alexandra Wood, Mark Bun, Marco Gaboardi, Urs Gasser, David R O'Brien, Thomas Steinke, and Salil Vadhan. 2017. Bridging the gap between computer science and legal approaches to privacy. *Harv. JL & Tech.* 31 (2017), 687.

[31] Laurent Sacharoff. 2018. Unlocking the Fifth Amendment: Passwords and Encrypted Devices. *Fordham Law Review* 87 (2018), 203–251. Issue 1.

[32] Sarah Scheffler and Mayank Varia. 2021. Protecting Cryptography Against Compelled Self-Incrimination. In *USENIX Security 2021*, Michael Bailey and Rachel Greenstadt (Eds.). USENIX Association, 591–608.

[33] Schmerber v. California, 384 U.S. 757. 1966.

[34] Sec. & Exch. Comm'n v. Huang, No. CV 15-269, 2015 WL 5611644 (E.D. Pa. Sept. 23). 2015.

[35] Seo v. State, 148 N.E.3d 952 (Ind.). 2020.

[36] State v. Andrews, 197 A. 3d 200 - NJ: Appellate Div. 2018.

[37] Dan Terzian. 2013. The Fifth Amendment, encryption, and the forgotten state interest. *UCLA L. Rev. Discourse* 61 (2013), 298.

[38] Dan Terzian. 2015. Forced Decryption as a Foregone Conclusion. *6 California Law Review Circuit 27* (2015).

[39] United States Constitution. Amendment V. 1791.

[40] United States v. Apple MacPro Computer, 851 F.3d 238 (3rd Cir.). 2017.

[41] United States v. Bright, 596 F. 3d 683 (9th Cir.). 2010.

[42] United States v. Burns, Dist. Court, MD North Carolina. 2019.

[43] United States v. Doe, 465 U.S. 605. 1984.

[44] United States v. Fricosu, 841 F. Supp. 2d 1232 (Dist. Court, D. Colorado). 2012.

[45] United States v. Greenfield, 831 F. 3d 106 (2nd Cir.). 2016.

[46] United States v. Hubbell, 530 U.S. 27. 2000.

[47] United States v. Kirschner, 823 F. Supp. 2d 665 - Eastern District of Michigan. 2010.

[48] United States v. Maffei, Dist. Court, ND California. 2019.

[49] United States v. Ponds, 454 F. 3d 313 (D.C. Cir.). 2006.

[50] John Henry Wigmore. 1961. A Treatise on the Anglo-American System of Evidence in Trials at Common Law; Including the statues and judicial decisions of all jurisdictions of the united states, John Theodore McNaughton (Ed.), Vol. 8.

[51] Andrew T. Winkler. 2013. Password Protection and Self-Incrimination: Applying the Fifth Amendment Privilege in the Technological Era. *Rutgers Computer & Technology Law Journal* 39 (2013), 194–215. Issue 2.

[52] Karen Yeung. 2019. Regulation by blockchain: the emerging battle for supremacy between the code of law and code as law. *The Modern Law Review* 82, 2 (2019), 207–239.

A MORE COMPELLED ACTS OF COMPUTATION

A.1 Two-factor authentication

Many services and devices lock via *two-factor authentication* (2FA) in which two "forms" of authentication are required. A typical setting is for a service to require entering a passcode and entering an ephemeral code that the service sent to a second device controlled by the same individual. In this section, we extend the example in Section 6.1 of decrypting a device by entering a password to require a code sent to a secondary device – the location of which may not be known to the government, although they must know the respondent can access it.

Claim A.1. $\mathcal{V}_{2\text{fa}}$ is demonstrable with respect to $\mathcal{E}_{2\text{fa}}$ and exemplar action $\mathcal{A}^*{}_{2\text{fa}}$.

Proof. To show demonstrability, we must show that $\mathcal{R}$ is capable of performing $\mathcal{A}^*{}_{2\text{fa}}$ and that $\mathcal{V}_{2\text{fa}}$ accepts $\mathcal{A}^*{}_{2\text{fa}}$ with respect to all $\mathcal{E}_{2\text{fa}}$-conforming $\mathcal{R}$ and $\mathcal{N}$. The first half is trivial since $\mathcal{A}^*{}_{2\text{fa}}$ only calls methods of $\mathcal{R}$ that were declared in $\mathcal{E}_{2\text{fa}}$.

To show the second half, first note that there is a device at location $\mathcal{N}[\text{deviceLoc}]$ that implements D. Call it M^D. We observe that since $\mathcal{E}_{2\text{fa}}$ declares that $\mathcal{R}.\text{pwd}() == D.pwd$, entering $\mathcal{R}.\text{pwd}()$ as input to $M^D.\text{promptPwd}$ is the same as entering $D.pwd$, and thus the call to promptPwd will set *code* and set *gotPwd* to true.

The evidence also states that there is a device at another location $\mathcal{N}[\mathcal{R}.\text{findSecond}()]$ implementing S. Call it M^S. By the last assertion in $\mathcal{E}_{2\text{fa}}$ we have that the call to $M^S.\text{getCode}()$ will yield a code c equal to $D.code$. Thus, entering that c into $M^D.\text{promptCode}$ will set *decrypted* to true (recall that *gotPwd* was set to true earlier).

	$k' = \mathcal{R}.k$	k' fixed in $\mathcal{T}$	k sampled in $\mathcal{T}$
$\mathcal{E}_{\text{secret}}$	NE (Thm. 5.4)	NE (Thm. 5.4)	NE (Thm. 5.5)
$\mathcal{E}_{\text{known}}$	NE (Thm. 5.4)	E (Rem. A.4)	NE (Thm. 5.5)

Table 1: Is $\mathcal{T} = \textbf{OTP}(k, \mathcal{R}.x)$ entailable (E) or not entailable (NE), for the given setting of $\mathcal{E}$ and generation of k?

Then when $\mathcal{V}_{\text{2fa}}$ calls M^D.read(), the device will return $m \neq \perp$, and thus $\mathcal{V}_{\text{2fa}}$ accepts as desired. □

Claim A.2. $\mathcal{V}_{\text{2fa}}$ entails $\mathcal{T}_{\text{2fa}}$ with respect to the stronger evidence $\mathcal{E}_{D,S} \succeq \mathcal{E}_{\text{2fa}}$ (i.e., that the specifications of D and S are full specifications).

Proof. Much like the proof given in Section 6.1, we will show that both running $\mathcal{P}_{\text{2fa}}$ on the transcript of $\langle \mathcal{V}_{\text{2fa}}^{\mathcal{N}}, \mathcal{A}^{\mathcal{R},\mathcal{N}} \rangle$, and running the target action $\mathcal{T}_{\text{2fa}}$, yield the same result of $D.m$.

For the left hand side, observe that if $\mathcal{V}_{\text{2fa}}$ returns Accept, then $m \leftarrow \mathcal{N}[\mathcal{E}.\text{deviceLoc}].\text{read}()$ was not $\perp$. $\mathcal{E}_{D,S}$ states that $\mathcal{N}[\mathcal{E}.\text{deviceLoc}]$ exactly implements D (and similarly for S). Hence, the only way for read() to return anything other than $\perp$ was for it to return $D.m$, which cannot be altered using any of the methods provided. Thus, if $\mathcal{V}_{\text{2fa}}$ returns Accept, the post-processor $\mathcal{P}_{\text{2fa}}$ must also return $D.m$.

Consider the right hand side. Because $\mathcal{E}_{D,S}$ specifies that $\mathcal{R}.\text{pwd}() = D.pwd$, this ensures that in the first line of $\mathcal{T}_{\text{2fa}}$, $x = D.pwd()$. Moreover, the second line of $\mathcal{T}_{\text{2fa}}$ calls exactly the given code of $D.\text{promptPwd}(D.pwd)$, which also calls exactly $\text{setCode}(D.code)$ no matter what the randomness tape of D is. The third line of $\mathcal{T}_{\text{2fa}}$ sets c to exactly $D.code$, and so the fourth line of $\mathcal{T}_{\text{2fa}}$ calls $D.\text{promptCode}(D.code)$. Thus, *decrypted* is set to true and when $\mathcal{T}_{\text{2fa}}$ calls read(), it must return $D.m$, as desired.

Thus, $\mathcal{V}_{\text{2fa}}$ entails $\mathcal{T}_{\text{2fa}}$ as desired. □

A.2 Preimage of hash

We imagine a scenario in which $\mathcal{G}$ wishes to compel $\mathcal{R}$ to provide a file and verify the file using its *hash*. This example is captured in Figure 4, where we envision h to be a fixed hash function like SHA-256. By construction of the evidence, $\mathcal{V}_{\text{hash}}$ is demonstrable. Entailment is more complicated.

Claim A.3. For all $\mathcal{E}_{\text{hash}}$-consistent $\mathcal{N}, \mathcal{R}$ and for all $\mathcal{V}_{\text{hash}}$-conforming $\mathcal{A}$, either

(i) $\mathcal{P}^{\mathcal{N}'}(\tau) = \mathcal{T}^{\mathcal{R},\mathcal{N}}$, or
(ii) $\mathcal{P}^{\mathcal{N}'}(\tau)$ and $T^{\mathcal{R},\mathcal{N}} = \mathcal{N}[\mathcal{R}.\text{findFile}()].\text{read}()$ are a hash collision for h.

Proof. Let $x_t = \mathcal{N}[\mathcal{R}.\text{findFile}()].\text{read}()$ be the output of $\mathcal{T}^{\mathcal{R},\mathcal{N}}$. The evidence ensures that $h(x_t) = \mathcal{E}.y$. Let x_p be the output of $\mathcal{P}^{\mathcal{N}'}(\tau)$. By definition, x_p is the message sent by $\mathcal{A}$ to $\mathcal{V}$. Because $\mathcal{A}$ is $\mathcal{V}$-conforming, $\mathcal{V}(x_p) = \text{Accept}$ and hence $h(x_p) = \mathcal{E}.y$. Hence $h(x_p) = h(x_t)$. Either $x_p = x_t$ or not, corresponding directly to the two cases in the claim. □

In words, this verifier *almost* entails $\mathcal{T}_{\mathcal{R},\mathcal{N}}$, except that the government has not ruled out the possibility that $\mathcal{R}$ knows a hash collision at y. Note that we have only shown that *this* verifier almost but not quite entails the action – a different verifier might very well entail the target action.

Since, in the real world, we do not expect the respondent to be capable of producing a hash collision, we expect that $\mathcal{R}$'s only strategy will be to send the exact file $\mathcal{G}$ wanted to entail. By compelling a preimage of a hash (which $\mathcal{G}$ knows with certainty $\mathcal{R}$ is capable of doing), this forces the respondent to either respond with the file or a hash collision, putting the ball in their court. Although the government did not have evidence that pinned down every degree of freedom the respondent had, $\mathcal{G}$ did not need to make any additional assumptions to create the demonstrable $\mathcal{V}$ shown, nor did $\mathcal{G}$ need to implement a potentially-more-costly verifier to find a different way of compelling the file that directly entailed the action. This demonstrates why the focus of our system is on *demonstrability* rather than entailment.

A.3 Compelling encryption and commitments

Throughout this section, we will use the following evidences: $\mathcal{E}_{\text{secret}}$ corresponds to the situation where $\mathcal{G}$ knows $\mathcal{R}$ has some secret, but has no knowledge of what it is and no way to verify it. $\mathcal{E}_{\text{known}}$ corresponds to a situation where $\mathcal{G}$ is able to learn the secret independently of $\mathcal{R}$ (by checking $\mathcal{N}[\text{xLoc}]$).

Evidence $\mathcal{E}_{\text{secret}}$:
Variables: $\mathcal{R}.x$, $\mathcal{R}.k$

Evidence $\mathcal{E}_{\text{known}}$:
Data: xLoc
Variables: $\mathcal{R}.x$, $\mathcal{R}.k$
Oracle (read only): $N[\text{xLoc}]$
assert: $\mathcal{R}.x == \mathcal{N}[\text{xLoc}].\text{read}()$

A.3.1 Compelling encryption. Let $\text{OTP}(k', m)$ return the bitwise XOR of k' and m assuming they are the same length. Table 1 shows whether there is any way of entailing the target action $\mathcal{T} = \text{OTP}(k', \mathcal{R}.x)$ depending on which key k' is used and on the evidence. Of the scenarios considered, $\mathcal{T}$ is entailable only when the government specifies a fixed key k' in $\mathcal{T}$ and is able to recover the plaintext $\mathcal{R}.x$ from nature.

Remark A.4. The action which deterministically chooses k' and then returns $\text{OTP}(k', \mathcal{R}.x)$ is entailable under $\mathcal{E}_{\text{known}}$. The verifier $\mathcal{V}$ that deterministically sets the same k and then sets $m \leftarrow \mathcal{N}[\text{xLoc}].\text{read}()$ is demonstrable and must always have exact equality with the output of this action, and so entails the action.

Remark A.5. Note that if the one-time-pad was replaced with a randomized encryption scheme, by Theorem 5.5 compelling the action of honestly encrypting m under the known k (using randomness in $\mathcal{T}$) is *not* entailable by Theorem 5.5. However, this action *is* entailable if $\mathcal{G}$ additionally fixes the randomness used in the encryption scheme, because $\mathcal{V}$ may check the exact equality of the result. This latter result stands in contrast with Theorem 4.5.1 from Scheffler-Varia [32], which found that compelling encryption under a freshly sampled key *was* compellable (under a different definition as described in Section 7).

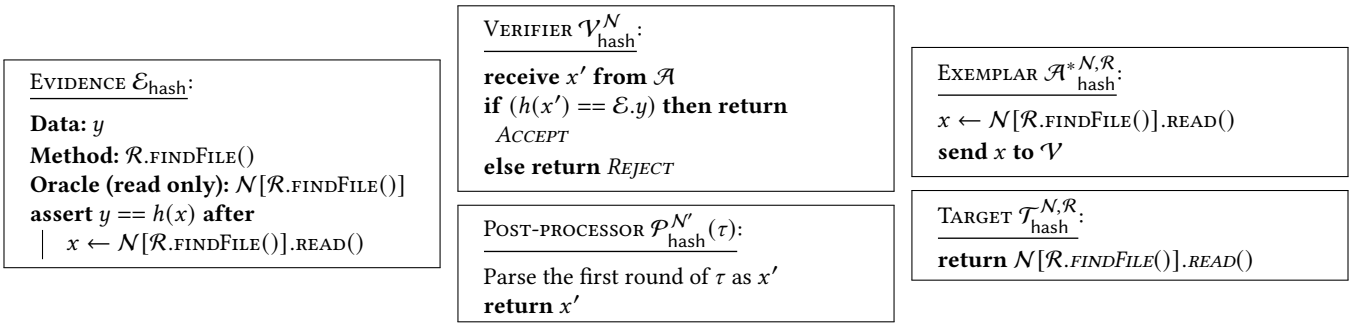

Figure 4: Verify a produced file using a hash (see Section A.2)

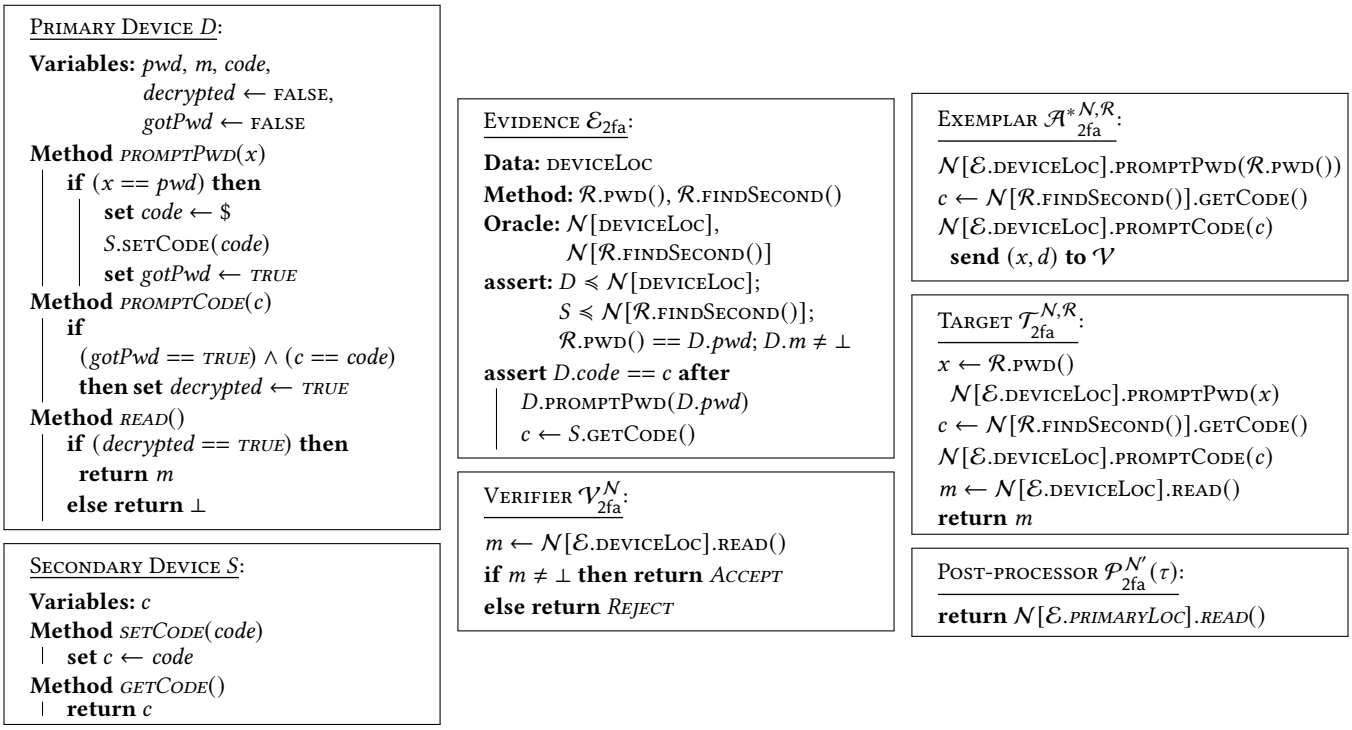

Figure 5: Multi-factor authentication (see Section A.1)

A.3.2 Compelling commitments. Together, the two claims in this section show that there is no way to entail a commitment of an unknown secret. Let (Com, Check) be a string commitment scheme as defined in Defn. 6.4. Recall $\mathcal{E}_{\text{secret}}$ above. Suppose $\mathcal{T}_{\text{com},\$}$ is the action which first sets $(c, d) \leftarrow \text{Com}_h(\mathcal{R}.x; \$)$ and then outputs c. That is, $\mathcal{T}_{\text{com}}$ outputs a fresh commitment to $\mathcal{R}.x$.

Claim A.6. There is no demonstrable $\mathcal{V}$ which entails $\mathcal{T}_{\text{com},\$}$ with respect to $\mathcal{E}_{\text{secret}}$.

Proof. Corollary of Theorem 5.5. □

Suppose instead that for a fixed string r, we consider the target action $\mathcal{T}_{\text{com},r}$ that first sets $(c, d) \leftarrow \text{Com}_h(\mathcal{R}.x; r)$, and then outputs c. This effectively "derandomizes" the $\mathcal{T}_{\text{com},\$}$ above.

Claim A.7. If $\forall(c', d') \in \text{Image}(\text{Com}), \exists x', \forall d''$:

$$(c', d'') \neq \text{Com}(x', r),$$

then there is no demonstrable $\mathcal{V}$ which entails $\mathcal{T}_{\text{com},r}$ with respect to $\mathcal{E}_{\text{secret}}$.

Proof. Corollary of Theorem 5.4. □

B PROOFS OF ENTAILMENT THEOREMS

B.1 Proof of Theorem 5.4

Proof. Suppose for contradiction $\exists \mathcal{E}' \geq \mathcal{E}_{\text{lang}}$, $\exists \mathcal{N}$ as in the hypothesis, and $\exists$ demonstrable $\mathcal{V}$ entailing $\mathcal{T}_{\text{lang}}$. Let $\mathcal{A}^*$ be the exemplar action guaranteed by demonstrability, and let $\mathcal{P}$ be the post-processor guaranteed by entailment.

We will construct $\mathcal{R}^*$ and $\mathcal{A}_0$ that violate the definition of entailment. We will have to show four things. First, $(\mathcal{N}, \mathcal{R}^*)$ are $\mathcal{E}'$-consistent. Second, that $\mathcal{A}_0$ is $\mathcal{V}$-conforming with respect to $(\mathcal{N}, \mathcal{R}^*)$. Third, that there is some set L^* such that $\mathcal{T}^{\mathcal{N},\mathcal{R}^*} \in L^*$. Fourth, that with with non-zero probability $\mathcal{P}^{\mathcal{N}'}(\tau) = x^* \notin L^*$. This contradicts our hypothesis, proving the theorem.

Let $\mathcal{R}_0 \in \mathfrak{R}_{\mathcal{N},\mathcal{E}'}$. Consider $\mathcal{A}_0$ the exemplar action corresponding to $\mathcal{V}$ with $\mathcal{R}_0$ hardcoded. That is, $\mathcal{A}_0$ emulates $\mathcal{A}^*$ but replaces any messages to $\mathcal{R}$ with a message to an emulated $\mathcal{R}_0$. By the definition of demonstrability, $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}_0^{\mathcal{N},\mathcal{R}} \rangle = \langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{*\mathcal{N},\mathcal{R}_0} \rangle$ returns Accept for all $\mathcal{R}$. Hence $\mathcal{A}_0$ is $\mathcal{V}$-conforming for all $\mathcal{R} \in \mathfrak{R}_{\mathcal{N},\mathcal{E}'}$.

Next we show that the distribution of $\mathcal{P}^{\mathcal{N}'}(\tau)$ is independent of $\mathcal{R}$ when $\mathcal{A} = \mathcal{A}_0$. For any $\mathcal{R}$, the execution $\langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}_0^{\mathcal{N},\mathcal{R}} \rangle = \langle \mathcal{V}^{\mathcal{N}}, \mathcal{A}^{*\mathcal{N},\mathcal{R}_0} \rangle$ is completely independent of $\mathcal{R}$. This is because $\mathcal{R}$ cannot interact with $\mathcal{N}$ or $\mathcal{V}$ directly, and $\mathcal{A}_0$ doesn't communicate with $\mathcal{R}$ by construction. Hence, the resulting state of nature $\mathcal{N}'$ and transcript τ are independent of $\mathcal{R}$, and hence $\mathcal{P}^{\mathcal{N}'}(\tau)$ is too.

Consider $x^* \in \mathsf{Support}(\mathcal{P}^{\mathcal{N}'}(\tau))$. By the theorem's hypothesis, $\exists \mathcal{R}^* \in \mathfrak{R}_{\mathcal{N},\mathcal{E}'}$ such that $x^* \notin L^*$, where $L^* \triangleq L_{\mathcal{R}^*,z}$. But $\mathcal{E}'$ implies that $\mathcal{T}^{\mathcal{N},\mathcal{R}^*} \in L^*$.

$\mathcal{R}^*$ and $\mathcal{A}_0$ violate the definition of entailment as required, completing the proof by contradiction. □

B.2 Proof of Theorem 5.5

Proof. Suppose by way of contradiction that there exists demonstrable $\mathcal{V}$ that entails $\mathcal{T}_{\mathsf{rand}}$. Let $\mathcal{A}^*$ be the exemplar action implied by demonstrability. Let $\mathcal{A}_0$ be $\mathcal{A}^*$ with the all-zeros randomness tape hardcoded. Because $\mathcal{V}$ is demonstrable, $\mathcal{V}$ must accept even when $\mathcal{A}^*$ is called with the all-zeros randomness tape, thus we know $\mathcal{A}_0$ is also $\mathcal{E}$-conforming.

Let $\mathcal{N}, \mathcal{R}$ have the property from the theorem statement, that is, there exists at least one setting of their randomness tapes for which $|\mathsf{Support}(\mathcal{T}_{\mathsf{rand}}^{\mathcal{N},\mathcal{R}})| \geq 2$. Fix their randomness tapes to the first such setting, we use $\mathcal{N}_0$ and $\mathcal{R}_0$ as shorthand for calling them with this specific randomness.

Now consider the entailment equation when considered with these fixed random tapes, that is, we require that for all settings of $\mathcal{P}$ and $\mathcal{T}_{\mathsf{rand}}$'s randomness, $\mathcal{P}^{\mathcal{N}_0'}(\langle \mathcal{V}^{\mathcal{N}_0}, \mathcal{A}_0^{\mathcal{R}_0,\mathcal{N}_0} \rangle) \equiv \mathcal{T}_{\mathsf{rand}}^{\mathcal{R}_0,\mathcal{N}_0}()$. (Note that there is no issue arising from the fact that the left hand side uses state $\mathcal{N}_0'$ after running $\langle \mathcal{V}^{\mathcal{N}_0}, \mathcal{A}_0^{\mathcal{R}_0,\mathcal{N}_0} \rangle$ rather than the original state $\mathcal{N}$; this argument will rely only on the fact that $\mathcal{N}_0$ is deterministic, not any other property of $\mathcal{N}_0$.) By assumption, the right hand side has support size at least 2.

If $\mathcal{P}$ is deterministic, the left hand side has support size 1. So for some setting of $\mathcal{T}_{\mathsf{rand}}$'s randomness, the two sides are not equal, which is a contradiction. If $\mathcal{P}$ is randomized, then fix $\mathcal{P}_0$ as $\mathcal{P}$ called with the all-zeros random string. Because our definition of entailment requires that the two sides be equal for all possible settings of $\mathcal{P}$ and $\mathcal{T}_{\mathsf{rand}}$'s randomness tape, the two sides must still be equal even for $\mathcal{P}_0$. However, once again, the left hand side now has support size 1, and the right hand side has support size at least 2. So there is at least one setting of $\mathcal{T}_{\mathsf{rand}}$'s randomness for which the two sides are not equal, which is a contradiction.

Thus, there is no demonstrable $\mathcal{V}$ that entails $\mathcal{T}_{\mathsf{rand}}$. □