The SSSCA: What does it mean for us and for our society?

Will Benton
willb@cs.wisc.edu

University of Wisconsin-Madison
Outline

- About me
- A few fundamental ideas
- What does the SSSCA propose?
- What are the implications of the SSSCA?
About me

Who am I?
grad student here in CS
also interested in philosophy, law, and civil liberties

What do I do?
I'm interested in languages, systems, and security
and techniques to help programmers develop safe and efficient programs
About me

Who am I?
- grad student here in CS
- also interested in philosophy, law, and civil liberties
About me

Who am I?
- grad student here in CS
- also interested in philosophy, law, and civil liberties

What do I do?
- I’m interested in languages, systems, and security
- and techniques to help programmers develop safe and efficient programs
About me

What else do I do?

I’m also interested in techniques to help people use untrusted programs safely.
About me

What else do I do?
- I’m also interested in techniques to help people use untrusted programs safely.

Why do I care?
- Technology creates opportunities: some “good”, some “bad”.
- It’s important that people use technology responsibly
- It’s important that legislators (and the public!) understand technology
Fundamental assumptions

We’re going to assume three things about computers:

- Computers operate on collections of generic bits – ones and zeroes. Those bits have no intrinsic meaning.

- Bits are inherently copyable.

- Any security scheme that depends on tamper-resistant hardware or software, or on using some kind of secret to restrict where someone can copy bits, is doomed to fail.
Bits are meaningless

Let’s look at a recipe:

Chocolate brownies

2 eggs
1 cup sugar
1/2 cup shortening
1/2 cup flour
1/4 cup cocoa powder
1 tsp. vanilla
1/2 tsp. salt

Combine ingredients until combined;

bake at 325 F until done
Bits are meaningless

Here’s the way that looks to the computer (each letter represents a pair of bytes, or 16 bits):

6843 636f 6c6f 7461 2065 7262 776f 696e 7365 3220 6520 6767 3b73 3120 6320 7075 7320 6775 7261 203b 2f31 2032 7563 2070 6875 6574 696e 676e 203b 2f31 2032 7563 2070 6c6f 3b72 3120 342f 6320 7073 7e2e 6176 696e 6c6c 3b61 3120 322f 7420 7073 7e2e 6173 746c 4320 6d6f 6962 656e 6920 676e 6574 7520 746e 6c69 6320 6d6f 6962 656e 3b64 6220 6b61 2065 7461 3320 3532 4620
Now another recipe:

**Chocolate brownies**

- 2 eggs
- 1 cup sugar
- 1/2 cup **shoe shine**
- 1/2 cup flour
- 1/4 cup cocoa powder
- 1 tsp. vanilla
- 1/2 tsp. salt

Combine ingredients until combined;

bake at 325 F until done
Bits are meaningless

A *very* different recipe – but how different are the bits?

| 6843 636f 6c6f 7461 2065 7262 776f 696e 7365 3220 6520 6767 3b73 3120 6320 7075 7261 203b 2f31 2032 7563 2070 6873 656f 7320 6968 656e 203b 2f31 2032 7563 2070 6c6f 3b72 3120 342f 6320 7073 207e2e 6176 696e 6c6c 3b61 3120 322f 7420 7073 2e 6173 746c 4320 6d6f 6962 656e 6920 676e 6572 6964 6e65 7374 7520 746e 6c69 6320 6d6f 6962 656e 3b64 6220 6b61 2065 7461 3320 3532 4620 |

The SSSCA: What does it mean for us and for our society? – p.9
Bits are meaningless

Certainly not as different as the end results of following those two recipes!

```
6843 636f 6c6f 7461 2065 7262 776f 696e 7365 3220 6520 6767 3b73 3120 6320 7073 7e2e 6176 696e 6c6f 7320 6965 7320 6965 7374 7520 746e 6c69 6320 6d6f 6962 656e 6920 676e 6572 6964 6e65 7374
6873 656f 7320 6965 7320 6965 7374 7520 746e 6c69 6320 6d6f 6962 656e 3b64 6220 6b61 2065 7461 3320 3532 4620
```
Why recipes?

- Computer programs are really just “recipes” for the computer; like the “bits” versions of the recipes, they require an external meaning.
- We can’t look at a recipe we haven’t made before and determine (for sure) whether or not the end result will be palatable.
- The computer is even worse off than we are, because its “ingredients” (data) are meaningless bits, too!
Consequences of bit meaninglessness

- Computers as “the Golem” – amoral, literalistic servants
- The Golem will only do what you tell it, even if it is bad
Consequences of bit meaninglessness

- Computers can’t prevent people from doing “Bad Things” – and this is a “Good Thing”
- Programs are speech; they are protected by the Constitution.
- Even a seemingly-bad program, say to guide a seal-clubbing robot, can have productive, legitimate applications, like ridding your yard of gophers.
One legit application for seal clubbing
Bits are copyable

Not only are bits copyable, but most computer systems depend on copying bits!

“Making bits uncopyable is like making water not wet” [1]

A classic blunder: DVDs are encrypted, but DVD pirates simply copy all the bits from the disc.

If bits were uncopyable, you wouldn’t be able to steal MP3s – or listen to them – even ones that you owned the copyright to.

[1] Security expert Bruce Schneier said this.
A brief digression: computer sound

Maybe my assertions don’t convince you. Let’s look at how playing that MP3 works:

- Bits (the file) are copied off of the disk into memory
- Those bits are copied and transformed in a process that uncompressed the audio
- The bits representing uncompressed audio are then copied “somewhere”
Somewhere?

Often, it’s a sound card, which does some additional things to the bits before converting them to an analog audio signal. But it could be:

- A file on the disk or across the network
- A printer
- Another program
- Any number of other “bit destinations”
Secrets are bad

Some computer systems will only let you run programs that have a special secret embedded in them, or copy data to a location that has a special secret associated with it.

This, however, creates a copyright oligarchy – the only people who can protect their copyrighted material are the ones who have access to the secrets!
Legislation is worse!

Current legislation in this arena (DMCA, AHRA) has served to hamper fair use and tools with legitimate uses, while enabling companies to use laughably weak encryption.

Would you want your bank to use weak encryption just because it was illegal to try and break it?
The SSSCA declares that the federal government must make a standard for “secure digital devices”.
The SSSCA makes a lot of things into federal felonies. Here are a few:

- manufacturing, selling, or importing “digital devices” that do not meet the gov’t spec
- reverse engineering or attempting to work around gov’t security for any reason
- telling others how to do same
What the SSSCA proposes

The SSSCA also

- exempts companies implementing “security standards” from antitrust law, at the discretion of the Secretary of Commerce

- allows the use of patented technology in “security standards” – meaning programmers can’t distribute free software

- effectively allows copyright holders to dictate the terms of their copyright
Are you kidding?

I wish I were. Here’s why:

- Under current copyright law, works enter the public domain after some period of time.
- There is no provision in the SSSCA (or the DMCA) for works to ever enter the public domain!
- Even worse, it’s *illegal* to circumvent protection *even* for works which are no longer copyrighted!
I bet it gets worse, right?

You bet it does. This law would make the Linux operating system, CS 537, and security research illegal. Why?

- The only way to control what a computer does is to modify it so that it will only run “special programs”

- However, Linux uses copyright in a different way; it is against the terms of its copyright to embed secrets in its code.

- Similarly, it would be illegal to develop programs that didn’t respect copy controls.
What about security?

Glad you asked:

- This law would make security worse
- Security is tested by people trying to break it, which would be illegal
- It’s already happened! See *Felten v. RIAA*
What about the future?

- Fair use and free speech are important; let’s not throw the baby out with the bathwater.
- Technology is good for the content industry; they just need a better business model!
- The real problem is that people have come to expect the ability to use copyrighted content illegally.