6.033 Spring 2017
Lecture #26

The Importance of Policy

or

Miscellaneous topics that you should know about and that we couldn't fit into the syllabus this semester.

net neutrality

2015 FCC ruling prohibits three things:

1. Blocking   ISPs can't block access to legal content, applications, services, or non-harmful devices

2. Throttling   ISPs can't impair or degrade traffic on the basis of content, applications, services, or non-harmful devices
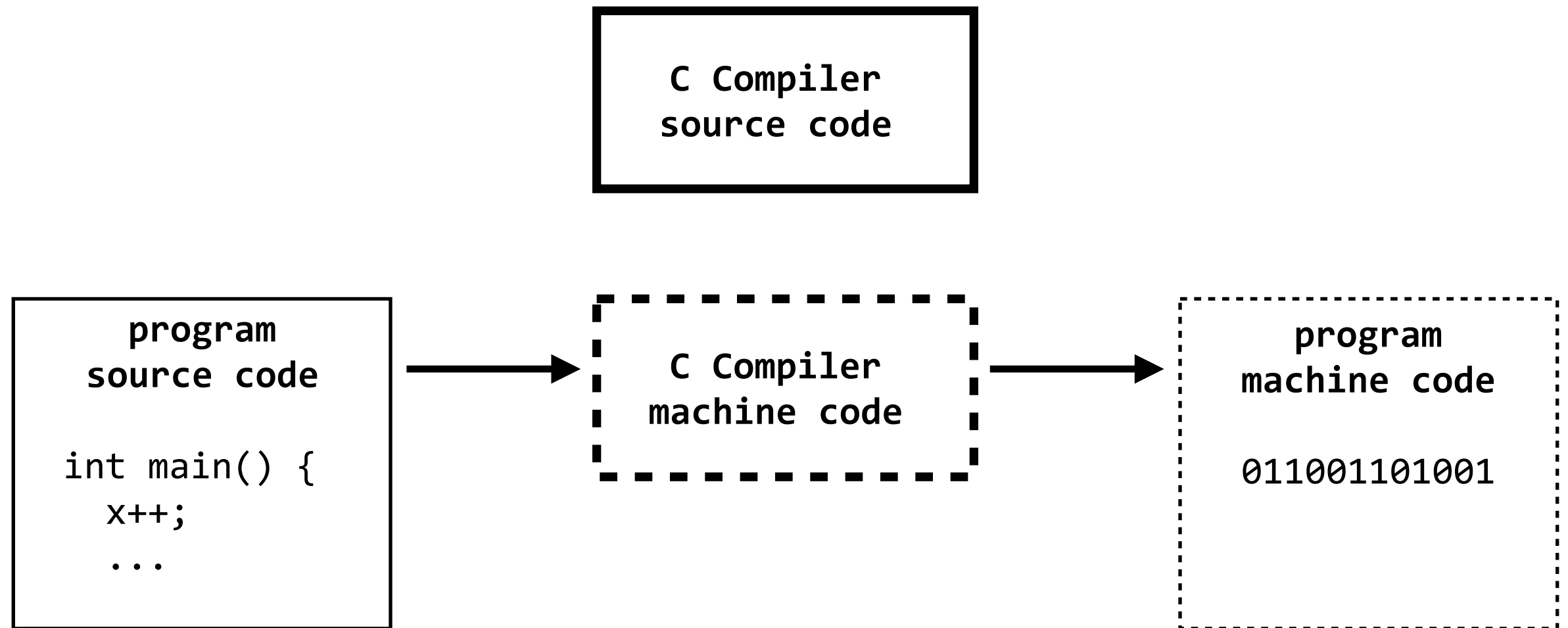
3. Paid prioritization   ISPs cannot favorite some lawful traffic over other lawful traffic in exchange for "consideration of any kind"

ISP *can* regulate the flow of traffic to and from **directly connected** networks

(Comcast can charge you more money for a faster connection to their network)

compilers: can we trust them?

C Compiler
source code

program
source code

int main() {
    x++;
    ...

C Compiler
machine code

program
machine code

011001101001

compilers: can we trust them?

compilers: can we trust them?

```
┌─────────────────┐        ┌╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌┐        ┌┄┄┄┄┄┄┄┄┄┄┄┄┄┄┄┐
│     UNIX        │───────▶│   C Compiler   │───────▶│     UNIX       │
│  source code    │        │  machine code  │        │  machine code  │
└─────────────────┘        └╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌┘        └┄┄┄┄┄┄┄┄┄┄┄┄┄┄┄┘
```

compilers: can we trust them?

```
┌─────────────────┐        ┌ ─ ─ ─ ─ ─ ─ ─ ─ ┐        ┌ · · · · · · · · ┐
│                 │        ┆                 ┆        ┆  Hacked UNIX    ┆
│  Hacked UNIX    │───────▶┆  C Compiler     ┆───────▶┆  machine code   ┆
│  source code    │        ┆  machine code   ┆        ┆  (has backdoor) ┆
│                 │        ┆                 ┆        ┆                 ┆
└─────────────────┘        └ ─ ─ ─ ─ ─ ─ ─ ─ ┘        └ · · · · · · · · ┘
```

This backdoor is easily discovered in the hacked UNIX source

compilers: can we trust them?

```
┌─────────────────┐          ┏━━━━━━━━━━━━━━━━━┓          ┌·················┐
│                 │          ┃     Hacked      ┃          ┊  Hacked UNIX    ┊
│     UNIX        │ ──────▶  ┃   C Compiler    ┃ ──────▶  ┊  machine code   ┊
│   source code   │          ┃  machine code   ┃          ┊  (has backdoor) ┊
│                 │          ┃                 ┃          ┊                 ┊
└─────────────────┘          ┗━━━━━━━━━━━━━━━━━┛          └·················┘
```
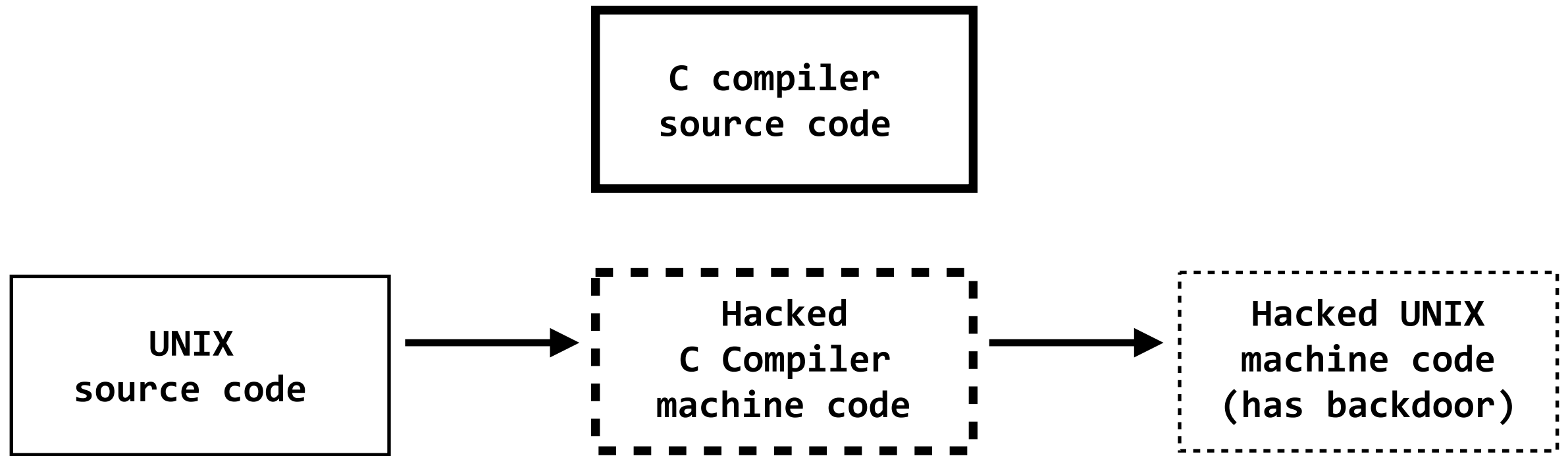
The hacked C compiler has code
that *inserts* a backdoor into UNIX

This backdoor *does not exist* in the UNIX source…

compilers: can we trust them?

```
┌─────────────────────┐
│      Hacked         │
│    C Compiler       │
│    source code      │
└─────────────────────┘
```

```
┌──────────────┐      ┌╌╌╌╌╌╌╌╌╌╌╌╌╌┐      ┌┄┄┄┄┄┄┄┄┄┄┄┄┄┄┄┐
│    UNIX      │      ┊   Hacked    ┊      ┊ Hacked UNIX   ┊
│ source code  │ ───► ┊ C Compiler  ┊ ───► ┊ machine code  ┊
│              │      ┊ machine code ┊     ┊ (has backdoor)┊
└──────────────┘      └╌╌╌╌╌╌╌╌╌╌╌╌╌┘      └┄┄┄┄┄┄┄┄┄┄┄┄┄┄┄┘
```
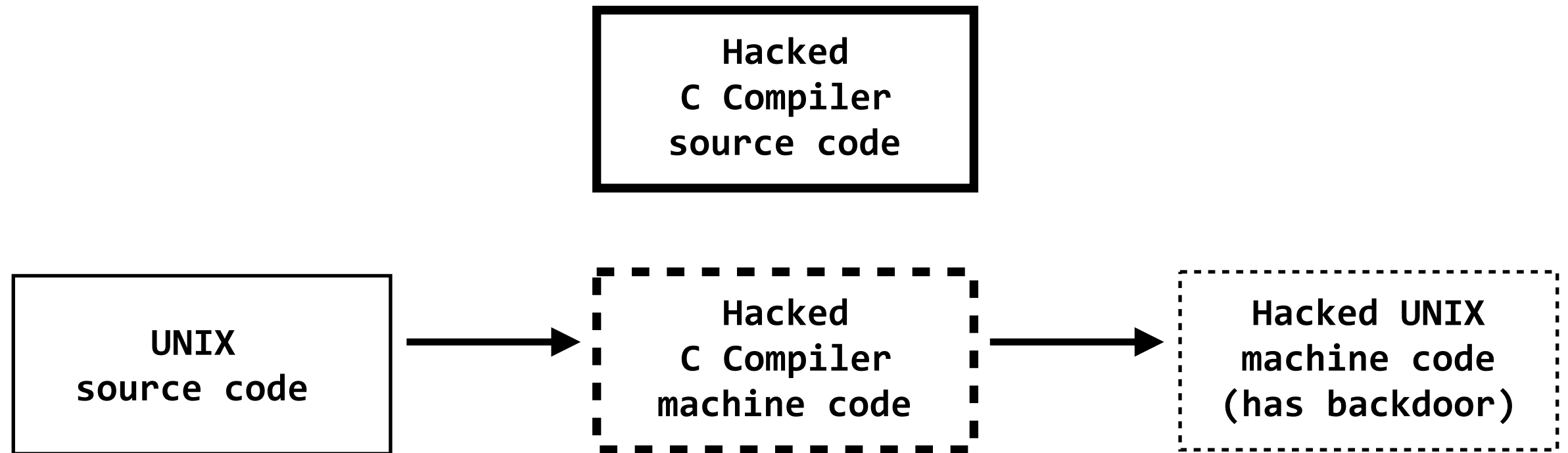
The hacked C compiler has code
that *inserts* a backdoor into UNIX

This backdoor *does not exist* in the UNIX source…

but it does exist in the hacked C Compiler source

compilers: can we trust them?


```
C compiler
source code
```

```
UNIX
source code
```

```
Hacked
C Compiler
machine code
```
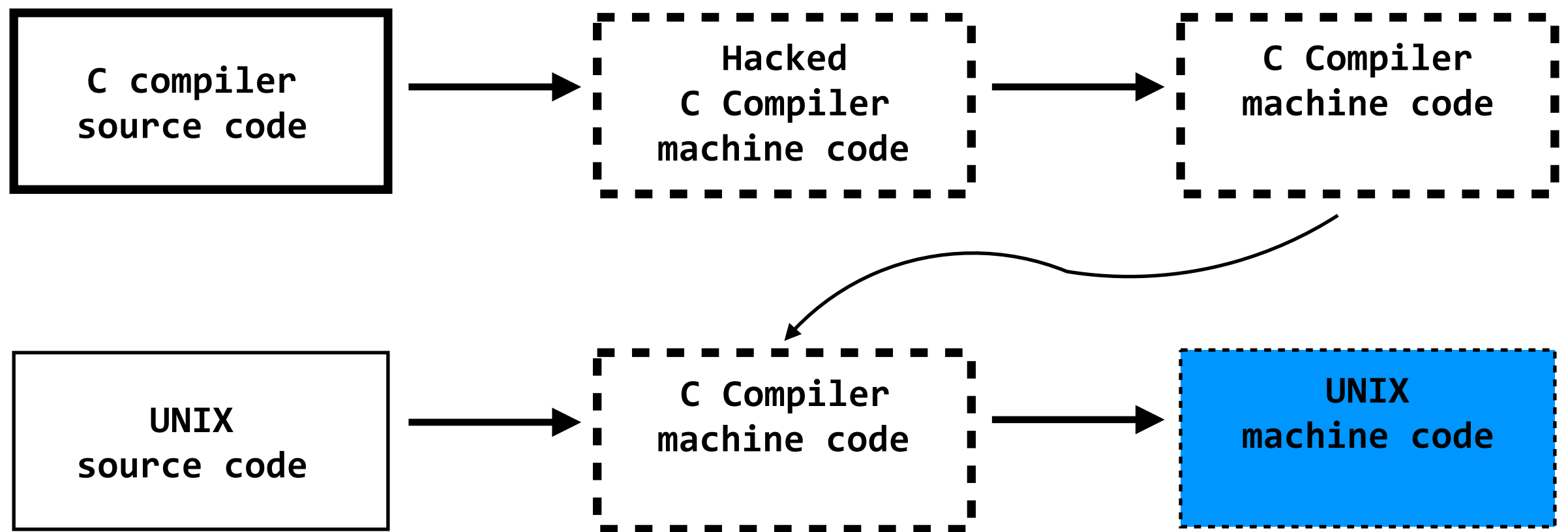
```
Hacked UNIX
machine code
(has backdoor)
```

The hacked C compiler has code
that *inserts* a backdoor into UNIX

What if I just lie, and tell you that the hacked C compiler was
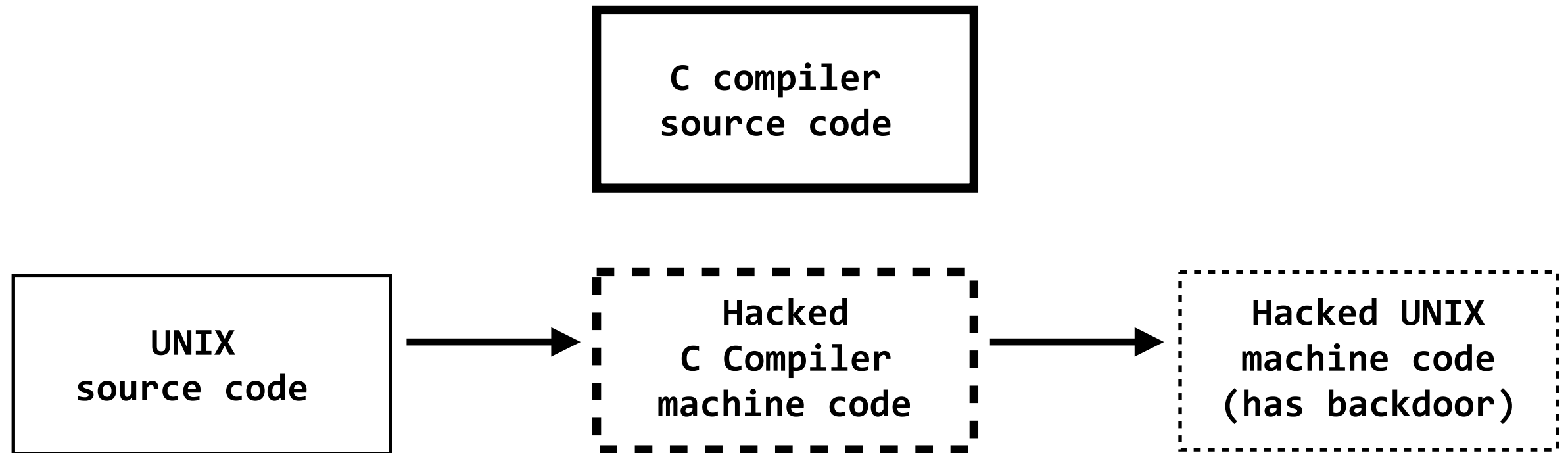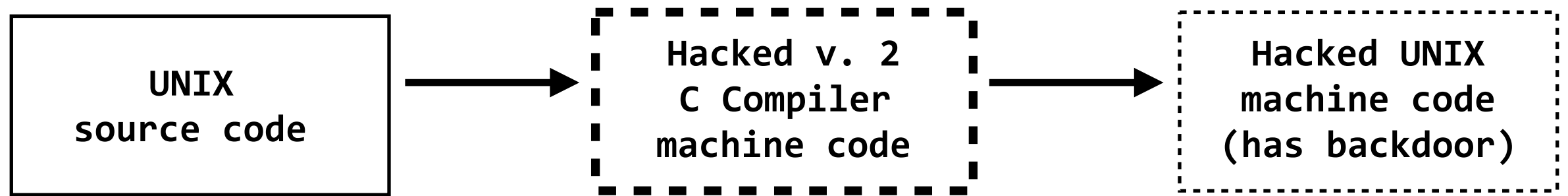generated from the clean C compiler source? Can you check?

compilers: can we trust them?

UNIX source code → Hacked C Compiler machine code → Hacked UNIX machine code (has backdoor)

The hacked C compiler has code that *inserts* a backdoor into UNIX

C compiler source code → Hacked C Compiler machine code → C Compiler machine code

UNIX source code → C Compiler machine code → UNIX machine code

compilers: can we trust them?

```
┌─────────────────────┐
│     C compiler      │
│    source code      │
└─────────────────────┘

┌──────────────┐      ┏━━━━━━━━━━━━━━━┓      ┌───────────────────┐
│    UNIX      │      ┃    Hacked     ┃      ┆  Hacked UNIX      ┆
│  source code │ ───▶ ┃  C Compiler   ┃ ───▶ ┆  machine code     ┆
│              │      ┃  machine code ┃      ┆  (has backdoor)   ┆
└──────────────┘      ┗━━━━━━━━━━━━━━━┛      └───────────────────┘
```

The hacked C compiler has code
that *inserts* a backdoor into UNIX

What if I just lie, and tell you that the hacked C compiler was
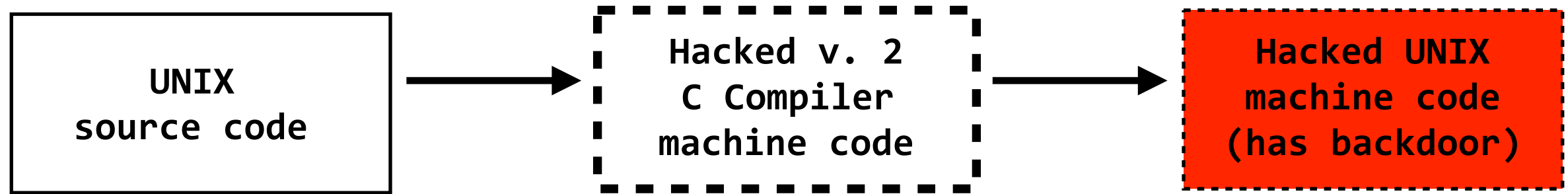generated from the clean C compiler source? Can you check?

Yes. By recompiling the compiler, and then compiling the
UNIX source

compilers: can we trust them?

```
┌─────────────────┐         ┏━━━━━━━━━━━━━━━━━┓         ┌┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈┐
│                 │         ┃   Hacked v. 2   ┃         ┊  Hacked UNIX  ┊
│     UNIX        │───────▶ ┃   C Compiler    ┃───────▶ ┊ machine code  ┊
│  source code    │         ┃  machine code   ┃         ┊ (has backdoor)┊
│                 │         ┗━━━━━━━━━━━━━━━━━┛         └┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈┘
└─────────────────┘
```
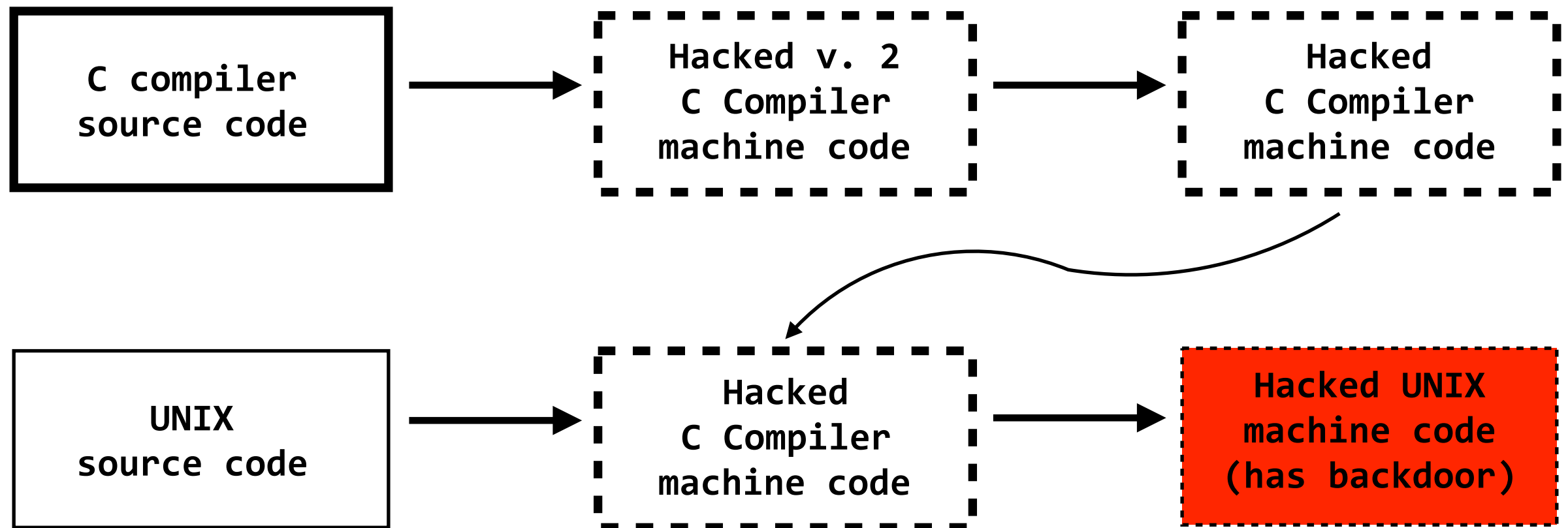
The hacked v. 2 C compiler has code
that *inserts* a backdoor into UNIX

and code to insert backdoor-
inserting code into C compilers

compilers: can we trust them?

| UNIX source code | → | Hacked v. 2 C Compiler machine code | → | Hacked UNIX machine code (has backdoor) |

The hacked C compiler has code that *inserts* a backdoor into UNIX *and* code to insert backdoor-inserting code into C compilers

| C compiler source code | → | Hacked v. 2 C Compiler machine code | → | Hacked C Compiler machine code |

| UNIX source code | → | Hacked C Compiler machine code | → | Hacked UNIX machine code (has backdoor) |

where to go from here

6.828 - Operating Systems
6.829 - Computer Networks
6.830/6.814 - Database Systems

6.858 - Computer Systems Security
6.857 - Network and Computer Security
6.875 - Cryptography and Cryptanalysis
6.892 - Shared Public Ledgers

more systems
↓
more math

6.824 - Distributed Systems ←— the most natural follow-up to 6.033
6.826 - Principles of Computer Systems
6.852 - Distributed Algorithms

6.903 - Intellectual Property
6.904 - Ethics for Engineers