

Trojan Source

Invisible Vulnerabilities

Nicholas Boucher
University of Cambridge

Ross Anderson
Universities of Cambridge & Edinburgh

Some Vulnerabilities are Invisible



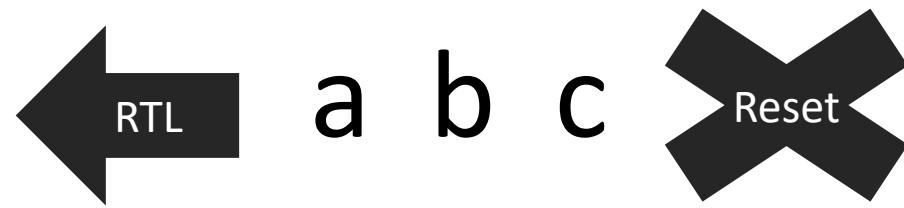
Encoding

a b c

Rendering

a b c

Encoding



Rendering

c b a

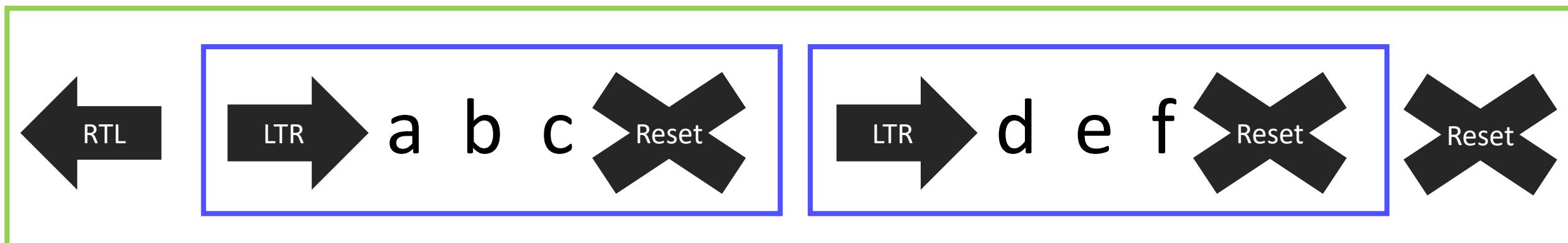
Encoding



Rendering

d e f a b c

Encoding



Rendering

d e f a b c

Source Code?

Comments + String Literals

The Vulnerability

1. **Control characters** can override text direction.
2. This can **modify display order**.
3. They can be **placed into comments and strings**.

⇒ Evil program A to be anagrammed into benign program B.

The Vulnerability

```
#include <stdio.h>
#include <stdbool.h>

int main() {
    bool isAdmin = false;
    /* begin admins only */ if (isAdmin) {
        printf("You are an admin.\n");
    /* end admins only */
    return 0;
}
```

```
$> |
```

The Vulnerability



Compiler Sees
Adversarial Logic

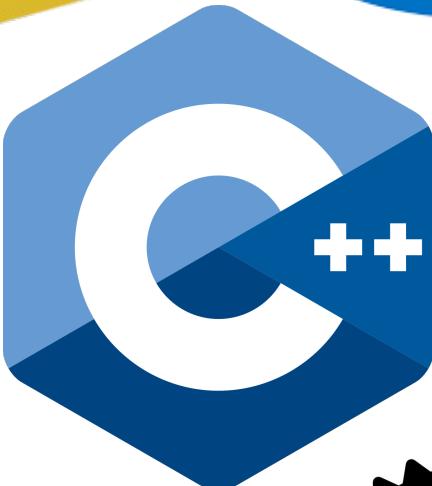
```
/* if (isAdmin) { begin admins only */
```

The Vulnerability

```
#include <stdio.h>
#include <stdbool.h>

int main() {
    bool isAdmin = false;
    /* begin admins only */ if (isAdmin) {
        printf("You are an admin.\n");
    /* end admins only */
    return 0;
}
```

```
$> clang program.c && ./a.out
You are an admin.
$> |
```



SOLIDITY



python

Supply Chain Attack

Coordinated Disclosure

GitHub

12 lines | 9 sloc | 228 Bytes

Raw Blame ⌂ ⌄ ⌅ ⌆

This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. [Learn more about bidirectional Unicode characters](#)

Hide revealed characters

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* [U+202E] If ([U+206A] [U+2069] [U+202E]) begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only [U+202E] ( [U+206A] [U+202E]) */
9
10 }
11
12
```

VS Code

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* [U+202E] } [U+2066]if (isAdmin) [U+2069] [U+2066] begin admins only */
7     |     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066]*/
9     return 0;
10 }
```

Bitbucket

```
trojan-source / commenting-out.c
```

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*<U202E> */ if (isAdmin) { begin admins only
7     printf("You are an admin.\n");
8     /* end admins only <U202E> */ }
9     return 0;
10 }
```

Rust

GitLab

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* begin admins only */ if (isAdmin) {
7         printf("You are an admin.\n");
8     /* end admins only */ }
9     return 0;
10 }
```

GCC

GitHub

12 lines (9 sloc) | 228 Bytes

Raw Blame   

 This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. [Learn more about bidirectional Unicode characters](#)

[Hide revealed characters](#)

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* U+202E } U+2066 if (isAdmin) U+2069 U+2066 begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only U+202E { U+2066 */
9     return 0;
10 }
```

JS
IT
TO }
@ LEGENDU 01
View raw file   

GitHub

12 lines | 9 sloc | 228 Bytes

Raw Blame ⌂ ⌄ ⌅

This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. [Learn more about bidirectional Unicode characters](#)

Hide revealed characters

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* [U+2020] [U+2066] If ([isAdmin]) [U+2069] [U+2066] begin admins only */
7     printf("You are an admin.\n");
8     /* [U+2020] [U+202E] ([U+2066]) */
9     /* end admins only */
10 }
11
12 }
```

VS Code

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] } [U+2066] if (isAdmin) [U+2069] [U+2066] begin admins only */
7     |     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066] */
9     return 0;
10 }
```

Bitbucket

```
trojan-source / commenting-out.c
```

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*<U202E> */ if (isAdmin) { // begin admins only
7         printf("You are an admin.\n");
8     /* end admins only */ } // <U202E> */
9     return 0;
10 }
```

Rust

GitLab

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* begin admins only */ if (isAdmin) {
7         printf("You are an admin.\n");
8     /* end admins only */ }
9     return 0;
10 }
11
```

GCC

VS Code

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] } [U+2066]if (isAdmin)[U+2069] [U+2066] begin admins only */
7     |     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066]*/
9     return 0;
10 }
11 }
12 return 0;
13 // Output: You are an admin.
```

GitHub

This screenshot shows a GitHub code editor window. The status bar at the top indicates "12 lines (9 sloc) | 228 Bytes". A yellow warning bar at the top states: "This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. Learn more about bidirectional Unicode characters". Below the bar is the actual code, which includes several bidirectional Unicode characters (U+202E and U+2066) and their corresponding visual representations.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* [U+202E] */ [U+2066]if (isAdmin) [U+2069] [U+2066] begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066]*/
9     return 0;
10 }
11
12 }
```

Bitbucket

This screenshot shows a Bitbucket code editor window. The status bar at the top indicates "12 lines (9 sloc) | 228 Bytes". A yellow warning bar at the top states: "This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. Learn more about bidirectional Unicode characters". Below the bar is the actual code, which includes several bidirectional Unicode characters (U+202E and U+2066) and their corresponding visual representations.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] */ <U2066>if (isAdmin)<U2069> <U2066> begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only <U202E> { <U2066>/
9     return 0;
10 }
11
12 }
```

GitLab

This screenshot shows a GitLab code editor window. The status bar at the top indicates "12 lines (9 sloc) | 228 Bytes". A yellow warning bar at the top states: "This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. Learn more about bidirectional Unicode characters". Below the bar is the actual code, which includes several bidirectional Unicode characters (U+202E and U+2066) and their corresponding visual representations.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] */ <U2066>if (isAdmin)<U2069> /* begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only <U202E> */ */
9     return 0;
10 }
11
12 }
```

VS Code

This screenshot shows a VS Code code editor window. The status bar at the top indicates "12 lines (9 sloc) | 228 Bytes". A yellow warning bar at the top states: "This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. Learn more about bidirectional Unicode characters". Below the bar is the actual code, which includes several bidirectional Unicode characters (U+202E and U+2066) and their corresponding visual representations.

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] */ [U+2066]if (isAdmin) [U+2069] [U+2066] begin admins only */
7     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066]*/
9     return 0;
10 }
11
12 }
```

Rust

This screenshot shows a terminal window displaying a warning message. The message is: "warning: unicode codepoint changing visible direction of text present in literal". It points to line 24 of a file named "src/test/ui/parser/unicode-control-codepoints.rs". The warning is triggered by the presence of bidirectional control characters (U+202E and U+2066) in a string literal. The terminal also shows some explanatory text about the nature of these characters and how they affect text flow.

```
24     println!("{}/* } if isAdmin begin admins only */",
```

note: `#[force-warn(text_direction_codepoint_in_literal)]` on by default
note: these kind of unicode codepoints change the way text flows on applications that support them, but can cause problems
help: if their presence wasn't intentional, you can remove them
24 println!("{}/* } /*\u{2066}if isAdmin\u{2069}\u{2066} begin admins only */",

GCC

This screenshot shows a terminal window displaying warnings from the GCC compiler. The warnings are related to unpaired UTF-8 bidirectional control characters. The first warning is at line 6 of "commenting-out.c", and the second is at line 8 of "commenting-out.c". Both warnings mention specific characters (U+202E, U+2066, U+202D, U+2067) and their context within the code. The terminal also shows the command used to run the compiler.

```
$ gcc -c trojan-source/C/commenting-out.c -fdiagnostics-escape-format=bytes
trojan-source/C/commenting-out.c: In function 'main':
trojan-source/C/commenting-out.c:6:43: warning: unpaired UTF-8 bidirectional control characters detected [-Wbidi-chars]
    /* begin admins only */ if (isAdmin)<U2069> <U2066> begin admins only */
    ^~~~~~
    |           |
    |           U+202E (RIGHT-TO-LEFT OVERRIDE)
    |           |
    |           U+2066 (LEFT-TO-RIGHT ISOLATE) end of
    |           bidirectional context
trojan-source/C/commenting-out.c:8:28: warning: unpaired UTF-8 bidirectional control characters detected [-Wbidi-chars]
    /* end admins only <U202E> */ */
    ^~~~~~
    |           |
    |           U+2066 (LEFT-TO-RIGHT ISOLATE)
    |           |
    |           end of bidirectional context
    |           U+2066 (LEFT-TO-RIGHT ISOLATE)
    |           |
    |           U+202E (RIGHT-TO-LEFT OVERRIDE)
```

Rust

GitHub

12 lines | 9 sloc | 228 Bytes

Raw Blame   

This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. [Learn more about bidirectional Unicode characters](#)

[Hide revealed characters](#)

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     bool isadmin = false;
7     /* [U+203E] [U+2066] if [isadmin] [U+2069] [U+2066] begin admins only */
8     printf("You are an admin.\n");
9     /* end admins only [U+202E] [U+2066] */
10    return 0;
11 }
12
```

VS Code

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*[U+202E] } [U+2066]if (isAdmin) [U+2069], [U+2066] begin admins only */
7     |     printf("You are an admin.\n");
8     /* end admins only [U+202E] { [U+2066]*/
9     return 0;
10 }
```

Bitbucket

```
trojan-source / commenting-out.c

1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /*<U202E> */ if (isAdmin) { /*<U2066> begin admins only */
7         printf("You are an admin.\n");
8     /* end admins only */ } /*<U202E> */ { /*<U2066>*/}
9     return 0;
10 }
11
12
13
14
15 }
```

Rust

GitLab

```
1 #include <stdio.h>
2 #include <stdbool.h>
3
4 int main() {
5     bool isAdmin = false;
6     /* begin admins only */ if (isAdmin) {
7         printf("You are an admin.\n");
8     } /* end admins only */
9     return 0;
10 }
```

GCC



About Unicode

Technical Quick Start Guide

Support Unicode

Adopt a Character

+

Become a Member

+

News and Information

+

Avoiding Source Code Spoofing

Unicode has convened a group of experts in programming languages, tooling, and security **to provide guidance and recommendations** on how to better handle international text in source code, as well as **providing code to help implementations**.

See paper for...

- Attack Generation
 - Attack Detection
 - Attack Variants
 - CVE Details
 - Compiler vs Editor vs Repo
 - GitHub Ecosystem Scanning
 - Coordinated Disclosure Details
 - Industry Response
 - Defenses
- ... and more

It's not just source code

- **LLMs** *Bad Characters: Imperceptible NLP Attacks*
(S&P 2022)
- **Search Engines** *Boosting Big Brother: Attacking Search Engines with Encodings*
(RAID 2023)
- **Coordinated Disclosure** *Talking Trojan: Analyzing an Industry-Wide Disclosure*
(SCORED 2022)

More Info



<https://trojansource.codes>