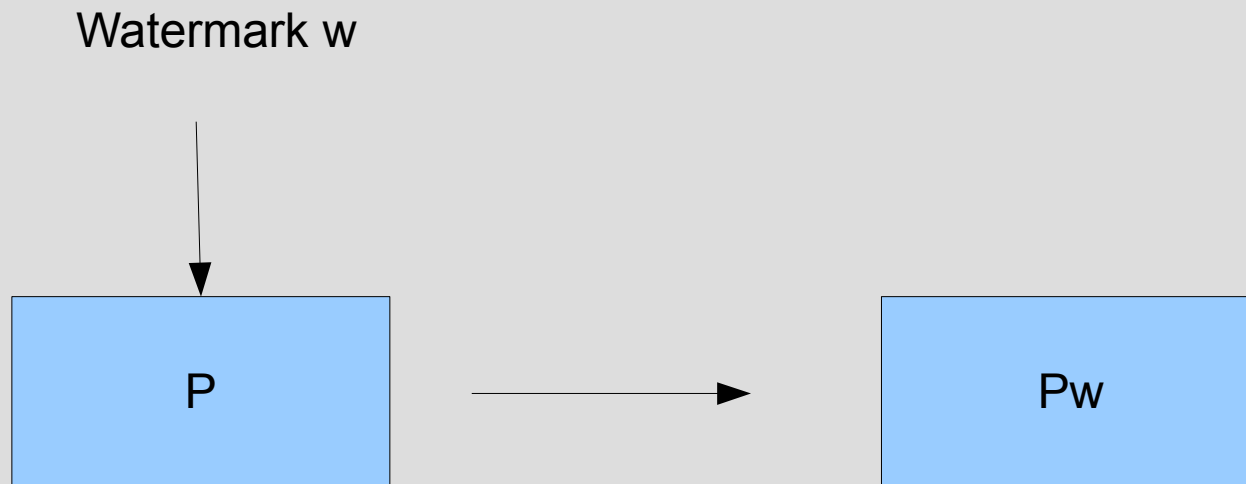


# Digital Watermarks for C/C++ Programs

Andrey Fionov

Siberian State University of Telecommunications  
and Information Science

# problem statement



Kinds of watermarks:

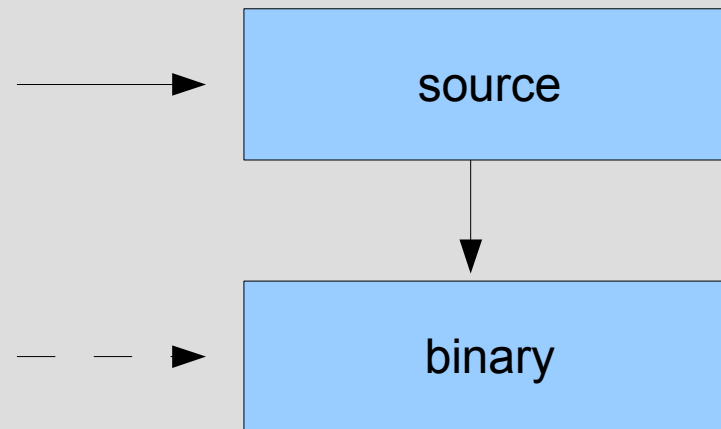
- Authenticating code
- Fingerprinting code

Requirements to watermarks:

- Hard to delete
- Hard to tamper

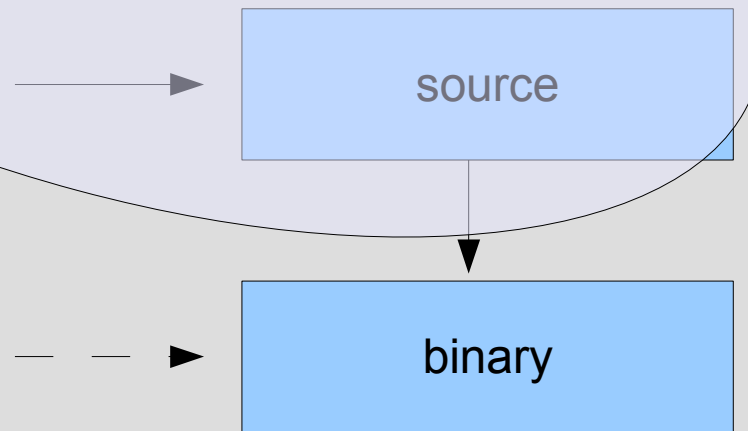
# two kinds of embedding

- Equivalent transformations
- Adding new code



# two kinds of embedding



- Equivalent transformations
- Adding new code





# example of equivalent transformation



```
int f (int x)
{
  int a, b;
  a = x + 1;
  b = x - 1;
  return a + 2 * b;
}
```

# example of equivalent transformation



```
int f (int x)
{
0  int a, b;
0  a = x + 1;
  b = x - 1;
  return a + 2 * b;
}
```

```
int f (int x)
{
1  int b, a;
1  b = x - 1;
  a = x + 1;
  return a + 2 * b;
}
```

# example of equivalent transformation

```
int f (int x)
{
0   int a, b;
0   a = x + 1;
  b = x - 1;
  return a + 2 * b;
}
```

```
x = [fp, #-16]
a = [fp, #-20]
b = [fp, #-24]
```

```
int f (int x)
{
1   int b, a;
1   b = x - 1;
  a = x + 1;
  return a + 2 * b;
}
```

```
x = [fp, #-16]
a = [fp, #-24]
b = [fp, #-20]
```

# example of equivalent transformation

```
ldr r3, [fp, #-16]
add r3, r3, #1
str r3, [fp, #-20]
ldr r3, [fp, #-16]
sub r3, r3, #1
str r3, [fp, #-24]
ldr r3, [fp, #-24]
mov r2, r3, asl #1
ldr r3, [fp, #-20]
add r3, r2, r3
mov r0, r3
```

```
ldr r3, [fp, #-16]
sub r3, r3, #1
str r3, [fp, #-20]
ldr r3, [fp, #-16]
add r3, r3, #1
str r3, [fp, #-24]
ldr r3, [fp, #-20]
mov r2, r3, asl #1
ldr r3, [fp, #-24]
add r3, r2, r3
mov r0, r3
```



# example of equivalent transformation

```
ldr r3, [fp, #-16]
add r3, r3, #1
str r3, [fp, #-20]
ldr r3, [fp, #-16]
sub r3, r3, #1
str r3, [fp, #-24]
ldr r3, [fp, #-24]
mov r2, r3, asl #1
ldr r3, [fp, #-20]
add r3, r2, r3
mov r0, r3
```

```
ldr r3, [fp, #-16]
sub r3, r3, #1
str r3, [fp, #-20]
ldr r3, [fp, #-16]
add r3, r3, #1
str r3, [fp, #-24]
ldr r3, [fp, #-20]
mov r2, r3, asl #1
ldr r3, [fp, #-24]
add r3, r2, r3
mov r0, r3
```

79:	203	103
91:	103	203
97:	30	24
105:	24	30

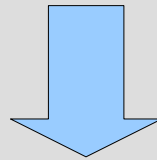
# some results

## Embedding Performance for Various SYMBIAN Projects

Project Name	The Number of C/C++ Files	The Size of C/C++ Files, bytes	The Length of Watermark, bits
HView v1 13beta source	10	58K	27
SymTorrent 1.30 source	104	680K	40
DosBox	107	2066K	123
putty src 1.5.1	227	3650K	1018
vim	395	3980K	207
OpenVideoHub	405	6853K	2205

# obstacles and future work

- compiler optimizations
- variables declaration in place of usage
- side effects of functions
- OOP
- indirections



design of steganographic code generator

THE END