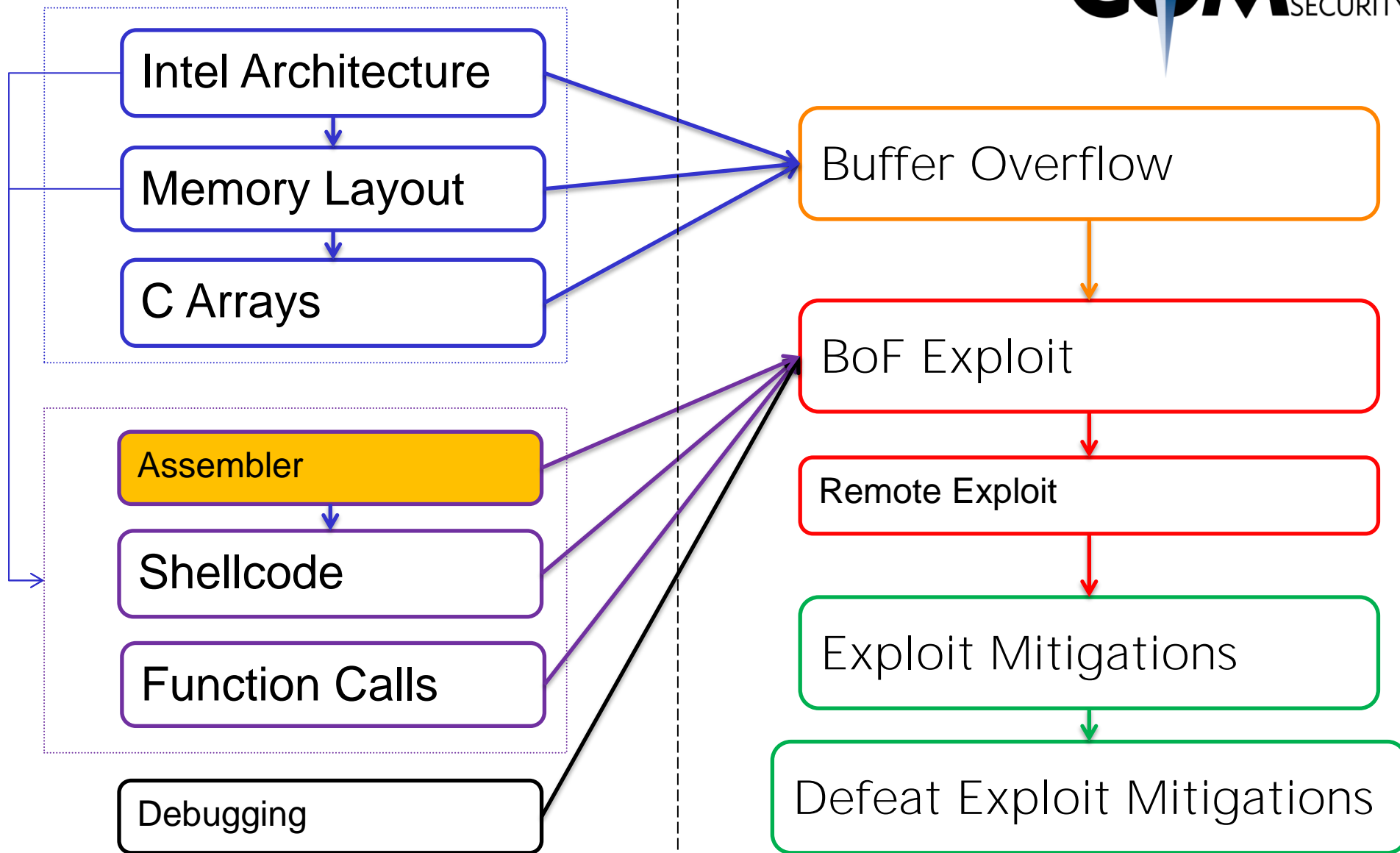




Assembler 101

Compass Security Schweiz AG
Werkstrasse 20
Postfach 2038
CH-8645 Jona

Tel +41 55 214 41 60
Fax +41 55 214 41 61
team@csnc.ch
www.csnc.ch





Short Assembler Intro

Compass Security Schweiz AG
Werkstrasse 20
Postfach 2038
CH-8645 Jona

Tel +41 55 214 41 60
Fax +41 55 214 41 61
team@csnc.ch
www.csnc.ch

Initialize a variable

```
int number;
```

```
number += 1
```

```
number dw 0
```

```
mov number, eax
```

```
inc eax
```

```
mov eax, number
```

Array on the stack

```
char test[5];
```

```
test[0] = 1;
```

```
test[3] = 9;
```

```
sub    $0x10,%esp
```

```
movb  $0x1,-0x5(%ebp)
```

```
movb  $0x9,-0x2(%ebp)
```

Array on the heap

```
char *test = malloc(5);
```

```
test[3] = 9;
```

```
sub    $0x28, %esp
movl   $0x5, (%esp)
call   8048300 <malloc@plt>
mov    %eax, -0xc(%ebp)
mov    -0xc(%ebp), %eax
add    $0x3, %eax
movb   $0x9, (%eax)
```

Conditional statement

```
if (number < 0) {
```

```
    <yes>
```

```
}
```

```
<no>
```

```
number dw 0
```

```
mov number, eax
```

```
cmp eax, 0
```

```
jge label    # jump greater equal
```

```
<yes>
```

```
label:
```

```
<no>
```

Loop

```
int n;
```

```
for(n=0; n<12; n++)
```

```
{
```

```
    printf("A");
```

```
}
```

```
+3   sub    $0x28,%esp
+6   movl   $0x0,-0xc(%ebp)
+13  jmp    0x8048403 <bla+31>
+15  movl   $0x41,(%esp)
+22  call   0x8048320 <putchar>
+27  addl   $0x1,-0xc(%ebp)
+31  cmpl   $0xb,-0xc(%ebp)
+35  jle    0x80483f3 <bla+15>
```