Lec8_App Security

Integer overflow

```
#include <stdio.h>
#include <limits.h>
int main(int argc, char* argv[]){
    int a;
    a = INT_MAX;
    printf("a=%d\n", a);
    a++;
    printf("New a=%d\n", a);
    return 0;
}
```

Integer underflow

```
#include <stdio.h>
#include <stdbool.h>
// Simulated function to check if the user has enough balance fo
bool has_sufficient_balance(unsigned int balance, unsigned int w
    // If withdrawal amount causes an overflow, this check may i
    if (balance - withdrawal_amount >= 0) {
        return true;
    }
    puts("Error: Withdrawal amount exceeds balance.");
    return false;
}
int main() {
    // User's balance
    unsigned int balance = 500;
    // The attacker's withdrawal amount, crafted to cause an int
```

```
    unsigned int attacker_withdrawal= 4294967295U;
    // Attempt the withdrawal
    if(has_sufficient_balance(balance, attacker_withdrawal)) {
        puts("Withdrawal approved.");
    } else{
        puts("Withdrawal denied.");
    }
    return 0;
}
```

Buffer overflow