**Introduction to cyber security  
Lesson 2: Strong passwords**

**Introduction**

In this lesson students explore the need for strong passwords and design an algorithm to create a strong password generator using the micro:bit.

**Time:** @60 minutes

**Learning objectives**

* To understand the need for secure password and what makes a password secure.
* To plan, test and debug an algorithm for a password generator.
* To use selection and variables in an algorithm and explain their use.

**Materials needed:** Lesson plan, lesson guide, rough paper, sticky notes and password generator planning sheets (printed or access on the school network).

**Lesson summary**

1. What is a password breach? (5 minutes)
2. Guess the password? (5 minutes)
3. Why are strong passwords important? (10 minutes)
4. Password Generator Task (10 minutes)
5. Writing an algorithm (20 minutes)
6. Testing and debugging algorithms (5 minutes)
7. Review & wrap up (5 minutes)

**Introduction: What is a password breach? (5 minutes)**

* Ask students to recap their understanding about malware from the previous lesson.
* Explain in this lesson you will be focusing on passwords and share the learning objectives on **slide 2** if you wish.
* Invite them to think/pair/share what is meant by a ‘password breach’ (**slide 3**) and discuss as a class.

**Guess the password (5 minutes)**

* Use **slide 4** to play a quick ‘Guess the password’ game.
* Invite students to share why it was so easy to guess and where hackers could find this kind of personal information (e.g. social media).
* Highlight the existence of powerful hacking software which helps hackers to guess passwords and that

people often use the same password for multiple accounts, inviting discussion about what problems this creates.

**Why are strong passwords important? (10 minutes)**

* Ask students to think/pair/share why strong passwords are important (**slide 5**).
* If you wish, give students a few minutes to look at the website on **slide 6** and check how safe Ali’s password and their own passwords are.
* Invite students to think/pair/share what makes a good/strong password (**slide 7**).

**Password generator challenge (10 minutes)**

* Explain to students that they are going to devise their own password generator using micro:bit and give them a few minutes to visit the website and answer the question on **slide 8**.
* Give sticky notes to individuals or pairs of students and ask them to devise their own secure password.
* Give students a few minutes to pair up and try to guess each other’s passwords. They should find it challenging and highlight that hackers or computer software would have the same problem.

**Writing algorithms (20 minutes +)**

* Discuss the process they went through to create their password (i.e. choose random numbers and letters, included capitals and symbols).
* Use **slide 9** to share the checklist reminder of what their password generator should be able to do.
* If helpful, use **slide 10** to show students a starter pseudocode algorithm. Check understanding by discussing what the next instruction(s) should be and if needed, **slides 11-14** can be used to help with selection and variables explanation.
* Ask students to complete the **password generator planning sheets** individually or in pairs, giving them a suitable amount of time and support to according to their confidence and experience.

**Testing and debugging algorithms (5 minutes)**

* Once students have completed their algorithm, ask them to share with another person or pair and work through them together to test and debug, ready to start coding in the next lesson (**slide 15**).

**Review and wrap up (5 minutes)**

* Ask students to recap their learning by answering the questions on **slide 16** and revisit the learning objectives if you wish on **slide 17**.

**Differentiation:**

**Support:**

* Give students the **password generator planning support sheet** to help them with their algorithm planning. They may also benefit from being given additional instructions that they can sequence and from being given adult support.

**Stretch & challenge:**

* Students explore the advice and resources for fostering good password practice offered by the UK National Cyber Security Centre <https://www.ncsc.gov.uk>
* Students could add a shake function to their algorithm with a range of characters such as hashtag, @ sign etc. to make their passwords more complex (see bottom of worksheet). They could also add sound effects to their algorithm.

**Opportunities for assessment:**

* Informal assessment of students’ answers to discussions.
* More formal assessment of students’ completed algorithms if wished.