

INSTRUCTOR GUIDE AND NOTES

Passwords and passphrases

A Take Back Your Bytes lesson built around a walkthrough of PrivacySafe Bot. A 50-minute lesson with a one-minute video, hands-on time with the live tool, and discussion.

Series	Take Back Your Bytes, PrivacySafe Foundation's cybersecurity education campaign ^[14]
Audience	Teens and adults. No technical background needed. Adapt the language for younger learners.
Length	50 minutes, with a 30-minute cut
Format	4 slides, a one-minute walkthrough video, hands-on time with the live tool, 4 slides, discussion
You will need	Slides (light for print, dark for projecting), the video on the lesson page, the live tool at privacysafe.bot , phones or calculators, flash cards
Facts checked	October 27, 2026. Guidance and tools change. Re-check before you teach.

PrivacySafe Foundation, Inc. | Take Back Your Bytes | These materials are dedicated to the public domain under CC0 1.0 and shared for the public good. The PrivacySafe and PrivacySafe Foundation names and logos are trademarks and are not included in that dedication. Names of other tools and organizations belong to their owners and are used as examples.

The lesson in five sentences

1. Secrets are guessed in several ways: lists of common passwords, reuse of leaked passwords on other sites, offline guessing after a breach, and phishing, which skips guessing^{[6][7]}.
2. Strength can be counted in bits. For random choices, bits equal length times $\log_2$ of the pool size, and each extra bit doubles the number of guesses^[3].
3. A random passphrase gains about 12.9 bits per word from a list of 7,776 words, so six words give about 77 to 78 bits^{[2][3]}.
4. Current guidance favors long secrets, allows long passwords, discourages forced symbols and scheduled changes, and recommends checking against known-breached passwords^[1].
5. Strength is not enough: use a unique secret for every account, a password manager and a second factor or a passkey, and prefer generators that run on your own device^{[8][9][10][11]}.

LEARNING OUTCOMES

By the end, students can:

1. Describe four ways passwords are attacked and which defense helps against each^{[6][7]}.
2. Estimate the strength of a random password or passphrase in bits^[3].
3. Explain why length and randomness matter more than clever substitutions.
4. Choose a secret type for a purpose: manager entry, master password or device login.
5. Write a short, realistic password plan.

Session plan

SEGMENT	MINUTES	NOTES
Before the video, slides 1 to 4	12	Includes the prediction: how many words equal 16 lowercase letters?
Walkthrough video	6	Play once with narration, then replay with learners calling out numbers.
Hands-on with the live tool	14	Generate, record bits, check one by hand.
After the video, slides 1 to 4	10	Passwords versus passphrases, the readout, beyond the password, plan.
Discussion	8	Two questions.

30-MINUTE CUT

Play the video once with narration and skip the second pass. Shorten the hands-on time to 8 minutes and do the by-hand check as a class. Cover after-video slides 1 and 3. Use one discussion question.

Before class

- Optional: the lesson page also has an overview video of under four minutes, in a light version for the page and a dark version for projecting, each with a subtitles file. It covers the whole lesson, and it is separate from the Bot clip. Check that both play on the room computer first.
- Play the video on the room computer.** It is about one minute long, has background music and no narration, and a caption file that identifies the music. Check the sound and the captions toggle.
- Open the live tool at privacysafe.bot^[12] and try both tabs. Its readout labels secrets weak, good or strong. Those tiers are the tool's own.
- Print or share the log2 table below so learners can check the math without special software.
- Plan for screens.** Anything you generate on a projector or shared screen is seen by the room. Tell learners not to use a secret generated in class for a real account.
- Check device policy for learners' phones and laptops. Learners should not change settings on school or work devices without permission.
- Print the flash cards, double-sided, flipped on the long edge.

LOG2 VALUES FOR THE BY-HAND CHECK

POOL	MEANING	LOG2
26	lowercase letters	4.70
52	upper and lowercase letters	5.70
62	letters and digits	5.95
7,776	words in the EFF long list	12.92

Phone calculators often lack log2. Use $\ln(\text{pool})$ divided by $\ln(2)$. Example: 16 lowercase letters are 16×4.70 , about 75 bits^[3].

TEACH IT FAIRLY

- Disclose the affiliation.** PrivacySafe Bot is built and hosted by the Foundation that publishes this lesson^[12]. Present it as one example. Password managers include generators, and EFF's dice method needs only dice and a list^[2]. Teach learners how to compare tools.
- The numbers are estimates.** They assume the secret was chosen at random. Human-made patterns are weaker.
- Avoid absolutes.** Guidance changes. Say what current guidance says and where learners can read it^[1].
- Do not shame.** Many people reuse passwords because the system is hard. Focus on a few doable steps.

ABOUT THE VIDEO

The video is a screen recording of the tool, about 63 seconds long, with background music and no narration. Its caption file contains one cue that identifies the music, "Mall" by Komiku, which is dedicated to the public domain under CC0 1.0^[13]. The lesson page also has a written description of what the video shows, for learners who cannot see or hear it. Use the narration script below when you present it.

Slide notes: before the video

Slide 1: Title

about 2 min

SAY

What makes a secret hard to guess, and why do the rules keep changing? Our promise: you will watch a real generator and see the math behind strength. Our job: learn what makes a secret strong, then build habits you can keep. The four-segment bar is the lesson: length, randomness, uniqueness, protection.

DO

Introduce Take Back Your Bytes as the Foundation's cybersecurity education campaign^[14].

CHECK FOR UNDERSTANDING

Ask: "What is the strongest password you have ever made? How did you make it?"

Slide 2: How passwords get guessed

about 4 min

SAY

- **Common passwords first:** attackers try popular and leaked passwords before anything else^[5].
- **Reuse:** a leaked password is tried on other sites, which is called credential stuffing^[7].
- **Offline guessing:** after a breach, stolen hashes can be guessed on the attacker's own machines, limited by their hardware and by how well the service stored them^[6].
- **Phishing:** a fake page tricks you into typing the password, so strength does not matter.

DO

Draw the four routes. Ask which defense matches each: strength, uniqueness, strength plus good storage, a second factor or a passkey.

CHECK FOR UNDERSTANDING

Ask: "If a site is breached and your password was 'Summer2024!', what happens next?"

WATCH FOR

Do not claim that a strong password makes you safe. It helps against guessing, not phishing.

Slide 3: Strength is a count of guesses

about 4 min

SAY

Strength is a count of how many guesses an attacker needs. For a random choice, bits equal length times $\log_2$ of the pool size, and each extra bit doubles the guesses^[3]. Walk the table: 8 lowercase letters are about 38 bits, 16 lowercase about 75, 16 characters from letters, numbers and symbols about 101, and words about 12.9 bits each, so 3, 4, 6 and 8 words give about 39, 52, 78 and 103 bits^[2]^[3].

DO

Work one line on the board, for example $8 \times 4.70 = 37.6$, then say "about 38 bits".

CHECK FOR UNDERSTANDING

Ask: "If I add one character to a lowercase password, how many times harder is it?" About 26 times, which is about 4.7 more bits.

WATCH FOR

The table assumes random choice. P@ssw0rd looks complex but is a known pattern, so it is far weaker than the math for random characters.

Slide 4: What to watch for, and the prediction

about 2 min

SAY

On the Password tab: the length slider, four character sets, the strength readout and its tier, and the bits estimate. On the Passphrase tab: the word count slider, the delimiter, how the bits change with each word, and where the tool says it runs.

DO

Read the prediction: "How many random words make a passphrase as strong as 16 random lowercase letters?" Take written guesses.

WATCH FOR

Learners often guess 3 or 4 words because words feel "short". The math says about 6.

Walkthrough video and narration (about 6 minutes)

The video is silent apart from music. Narrate it live. Times are approximate and match the copy supplied with this lesson. Pause on any readout you want to discuss.

TIME	WHAT IS ON SCREEN	SAY
0:00	Password tab, length 16, lowercase only	Here is PrivacySafe Bot in a browser. 16 lowercase letters give about 75 bits, in its strong tier.
0:04	Uppercase added	A bigger pool makes each character worth more: about 91 bits.
0:08	Numbers and symbols added	Now about 101 bits.
0:12	Length 30, then 32, all sets	Longer is much stronger: about 189 and 201 bits. Length adds more than variety.
0:20	Lowercase only, length 32, then 17	32 lowercase letters are still about 150 bits, more than 16 characters of every kind. Each extra letter adds about 4.7 bits.
0:28	Length 8, lowercase	8 lowercase letters are about 38 bits, and the tool labels them weak.

0:33	Passphrase tab, 6 words	Now random words. Six words are about 78 bits, close to the 16 lowercase letters. That answers the prediction.
0:40	Delimiter options	The delimiter changes how the words are joined, such as a hyphen, a space or a period.
0:48	8 words, space	Eight words are about 103 bits.
0:56	4 words, then 3 words	Four words are about 52 bits and labeled good. Three words are about 39 bits and labeled weak.

USE THE SAME SETTINGS FOR THE MATH

Use the hyphen, space and period delimiters for worked examples. The digit delimiter changes the shape of a passphrase, so explore it after learners understand the basic math.

Hands-on with the live tool (about 14 minutes)

1. Open privacysafe.bot^[12]. Generate three secrets: 12 lowercase letters, 16 characters with every character set, and a 6-word passphrase. Write down each length, the sets, and the bits shown.
2. Pick one secret and check it by hand: bits = length x log2(pool). Use the log2 table. Compare with the tool.
3. Find the settings where the tool changes tier. What is the smallest word count labeled good? Strong?
4. Test the prediction: how many words match 16 lowercase letters? Compare 6 words with 16 letters, then try 7 letters and 5 words.
5. Decide where each secret would live: a password manager entry, a master password, or a device login.

SAFETY REMINDER

Anything generated in class is visible to others nearby. Do not use a secret you generated in class for a real account. If you want a real secret, generate it privately and store it in a password manager.

Troubleshooting

SYMPTOM	LIKELY CAUSE AND FIX
The tool shows a blank page	It needs JavaScript. Check that scripts are allowed for the site.
Numbers differ by a bit or two	The tool rounds, and log2 values here are rounded. The by-hand answer should be within a bit or two.
Tier labels differ from the slides	The labels are the tool's own and may change. Teach the bits, then the label.
A phone calculator has no log2	Use $\ln(\text{pool})$ divided by $\ln(2)$, or use the table.
The video will not play	Check the file and sound. If it still fails, run the tool live and narrate the same steps.

Slide notes: after the video

Slide 1: Passwords versus passphrases

about 3 min

SAY

	PASSWORD	PASSPHRASE
What it is	Random characters	Random words
Remember it?	Not really	Much easier
Best for	Accounts kept in a manager	Master password, device login, Wi-Fi
Strength grows by	About 4.7 to 6.3 bits per character	About 12.9 bits per word ^[3]

DO

EFF suggests six words from its long list for most uses^[2]. Both kinds must be random, so use a generator or dice, not your own picks.

CHECK FOR UNDERSTANDING

Ask: "Why is a passphrase easier to type on a phone?" Fewer odd characters, and words are familiar.

WATCH FOR

A famous quote or song line is not a random passphrase. It is a pattern an attacker can guess.

Slide 2: Reading the strength readout

about 3 min

SAY

In the video the tool shows weak at about 38 and 39 bits, good at about 52 bits and strong from about 75 bits up. These tiers are the tool's own labels. Aim higher for secrets that protect a lot. A common recommendation is 16 or more random characters for accounts and 6 or more random words for a master password^{[2][8]}. NIST's current guidance for services asks them to allow long passwords, avoid forced symbols and scheduled changes, and check passwords against known breaches^[1].

CHECK FOR UNDERSTANDING

Ask: "Why do some sites still force a symbol and a change every 90 days?" Old habits, and rules that are hard to unlearn.

WATCH FOR

The slide separates advice for you from advice for sites. Keep it separate in discussion. Learners cannot change a site's rules, but they can choose strong secrets and complain.

Slide 3: Beyond the password

about 4 min

SAY

- **Unique for every account:** reuse lets one breach open many doors^[7].

- **A password manager:** it remembers long random secrets so you remember one^{[2][4][8]}.
- **A second factor or a passkey:** a stolen password is not enough^{[9][10]}.
- **Check for breaches:** Have I Been Pwned can show whether a password has leaked^[5].

Where a generator runs matters. PrivacySafe Bot describes itself as running in your browser on your device, and browsers provide a secure random source for this^{[11][12]}.

DO

Ask learners what they check before trusting a generator: where it runs, whether the source is public, who runs it, and what its privacy policy says.

WATCH FOR

Do not tell learners any one tool is the only safe choice. Teach the questions.

Slide 4: Your password plan

about 4 min

SAY

1. Use a password manager.
2. Make every password unique.
3. Use 16 or more random characters for accounts.
4. Use a 6-word random passphrase for your manager and devices.
5. Turn on a second factor or passkeys.
6. Never build secrets from personal facts or patterns.
7. Check whether accounts appear in known breaches.
8. Change a password after a breach, not on a schedule.

DO

Ask learners to pick the first step they will do this week.

Discussion (about 8 minutes)

QUESTION 1: WHY ARE THE RULES STILL STRANGE?

"Why do some sites still demand symbols and monthly changes? What would you change about their rules?"

- Old rules were written before research showed they push people toward patterns like adding "1!" at the end^[1].
- Current guidance favors length, allows long passwords and recommends breach checks^[1].
- Learners can tell sites about weak rules, and can choose services with better ones.

QUESTION 2: WHERE SHOULD A GENERATOR RUN?

"Where should a password generator run, on your device or on a server? What would you check before trusting one?"

- A generator that runs in your browser does not need to send the secret anywhere, and the browser provides secure randomness^[11].
- Check where it runs, whether the source is public, who operates it and what its policy says.
- A password manager's built-in generator is also an option.

Answer key and assessment

QUESTION	ANSWER
Name four ways passwords are attacked.	Common passwords, reuse, offline guessing after a breach, phishing ^[6] ^[7] .
How do you estimate bits for a random password?	Length times log2 of the pool size ^[3] .
What does each extra bit do?	Doubles the number of guesses ^[3] .
How many bits per word from a 7,776-word list?	About 12.9 ^[3] .
How many words match 16 lowercase letters?	About 6: about 78 bits versus about 75 bits ^[2] .
Why does a second factor help?	A stolen or guessed password alone is not enough ^[9] .

EXIT TICKET (3 MINUTES)

1. Which is stronger: 8 random characters from all four sets, or 4 random words? Show your math.
2. Why is a random passphrase better than a quote?
3. Name one step you will take this week.

HOMEWORK

- Set up or review a password manager, and replace one reused password with a unique random one.
- Turn on a second factor or a passkey for your email account.
- Stretch: look up the EFF dice method^[2] and generate a six-word passphrase with real dice.

Common questions and misconceptions

QUESTION OR BELIEF	RESPONSE
"Symbols make a password strong."	Length and randomness matter more. A short password with a symbol is still short ^{[1][3]} .
"I should change my passwords every month."	Current guidance says change one when there is a reason, such as a breach ^[1] .
"A long quote makes a great passphrase."	Only random words work. A known quote is a pattern.
"My strong password protects me from phishing."	No. A fake page can capture any password. Use a second factor or a passkey ^[9] ^[10] .
"If a tool says strong, I am done."	The label is an estimate that assumes random choice. Also use unique passwords and a second factor.
"Online generators are all unsafe."	It depends on where it runs and who operates it. A generator that runs on your device and shows its source is easier to trust. A password manager's generator is another option ^[11] .

References and further reading

Citations in this guide use bracketed numbers. Links were checked on October 27, 2026. Guidance and tools change, so confirm details before you teach. The video and its caption file are part of this lesson package.

#	SOURCE	WHAT IT SUPPORTS
[1]	NIST, SP 800-63B-4, Digital Identity Guidelines: Authentication and Authenticator Management (2025) https://pages.nist.gov/800-63-4/sp800-63b.html	Current US guidance on memorized secrets: favor length, allow long passwords, avoid composition rules and scheduled changes, and check against known-compromised passwords.
[2]	Electronic Frontier Foundation, EFF Dice-Generated Passphrases https://www.eff.org/dice	The long wordlist of 7,776 words, the advice to use six words, and the 77-bit figure.
[3]	Electronic Frontier Foundation, Deep Dive: EFF's New Wordlists for Random Passphrases (2016) https://www.eff.org/deeplinks/2016/07/new-wordlists-random-passphrases	How passphrase strength is measured: about 12.9 bits per word, and each extra bit doubles the guesses.
[4]	EFF Surveillance Self-Defense, Creating Strong Passwords https://ssd.eff.org/module/creating-strong-passwords	Plain-language advice on strong passwords and password managers.
[5]	Have I Been Pwned, Pwned Passwords https://haveibeenpwned.com/Passwords	A service for checking whether a password appears in known breaches.
[6]	OWASP, Password Storage Cheat Sheet https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html	How services should store passwords as salted hashes, and why stolen hashes can be guessed offline.
[7]	OWASP, Credential Stuffing Prevention Cheat Sheet https://cheatsheetseries.owasp.org/cheatsheets/Credential_Stuffing_Prevention_Cheat_Sheet.html	How attackers reuse leaked credentials on other sites, and defenses including multi-factor authentication.
[8]	CISA, Secure Our World: Use Strong Passwords https://www.cisa.gov/secure-our-world/use-strong-passwords	Consumer advice on long, random, unique passwords and password managers.
[9]	CISA, Secure Our World: Turn on MFA https://www.cisa.gov/secure-our-world/turn-mfa	Consumer advice on multi-factor authentication.
[10]	FIDO Alliance, Passkeys https://fidoalliance.org/passkeys/	An overview of passkeys, a phishing-resistant way to sign in.

[11]	MDN Web Docs, <code>Crypto.getRandomValues()</code> https://developer.mozilla.org/en-US/docs/Web/API/Crypto/getRandomValues	The browser's cryptographically secure random number source.
[12]	PrivacySafe Bot https://privacysafe.bot	The password and passphrase generator shown in the video. Its own pages describe how it runs and what it estimates.
[13]	Music credit in the video's caption file <code>video/privacysafe-bot-demo_en.vtt</code>	The background music, "Mall" by Komiku, is dedicated to the public domain under CC0 1.0.
[14]	PrivacySafe Foundation https://privacysafe.foundation/	The Take Back Your Bytes education campaign and the Foundation's other work.
[15]	Creative Commons, CC0 1.0 Universal https://creativecommons.org/publicdomain/zero/1.0/	The public-domain dedication that covers these lesson materials.

ABOUT THESE MATERIALS

The slides, video page, flash cards and this guide were written for the PrivacySafe Foundation's Take Back Your Bytes campaign. They are dedicated to the public domain under CC0 1.0^[15]. The Foundation names and logos are trademarks and are not included in that dedication. PrivacySafe Bot is built and hosted by the Foundation. This lesson is independent of the other organizations and tools named in the references.