Solutions

Each answer shows the steps, then the **result**.

2.1

- collision resistant - hard to find two inputs producing the same output; pre-image resistant - given a hash, infeasible to determine the input; repeatable - the same input always produces the same hash; fixed length - the output length is constant regardless of input size

2.2

- an n-bit hash has only 2^n possible outputs while the number of possible inputs is infinite, so two inputs must eventually share an output. It remains secure because the property claimed is not that collisions do not exist, but that **finding** one is infeasible; the function is broken only when an efficient algorithm to force a collision is found

2.3

- if the adversary gains access to the user:password directory, plaintext hands them every password immediately; pre-image resistance means a hash cannot feasibly be turned back into the password that produced it

2.4

(a)

- identical

(b)

- a salt adds a few random bits to each password before hashing, so the two stored hashes differ, which defeats a rainbow-table attack because a precomputed table of common passwords and their hashes no longer matches, and each password must be attacked separately

2.5

(a)

- online; detectable - it appears in the authentication log as many accounts each tried once from one IP

(b)

- offline; not detectable, because the work happens on the adversary's own computer against a captured hash and touches no system you own

2.6

- switches, routers and IoT devices are preconfigured with a default administrative user and password, and these defaults are published, so trying a short list of `user:password` pairs succeeds on any device where the default was never changed

2.7

(a)

- biometric

(b)

- possession

(c)

- knowledge

(d)

- location

2.8

- both are **knowledge** factors - things the user knows; MFA requires more than one **type** of factor, and an adversary who phished the password can usually also find the answer to a challenge question, so no independent layer has been added

2.9

(a)

- password history

(b)

- lockout after a number of invalid attempts

(c)

- complexity, and minimum length

(d)

- maximum password age

3.1 C

- 2^n outputs against infinitely many inputs, which is why collisions must exist

3.2 B

- It does not make the hash reversible or irreversible; it removes the equality between identical passwords

3.3 D

- The offline attack runs on the adversary's computer, so it produces no entry in any log the organisation holds

3.4 B

- One password per account keeps every account below its lockout threshold

3.5

(a)

- an **offline** hash-cracking attack, most likely using a rainbow table. The cracking took place on the adversary's own computer against the copied hash file, so by the time they logged in they already had the correct passwords - hence every login succeeded first time and no failed attempts were ever recorded

(b)

- without a salt, two identical passwords hash identically, so a **rainbow table** - a precomputed list of common passwords with their hashes - can be matched against the whole directory at once, turning cracking into a lookup; and one cracked hash immediately reveals every other account using that password, so forty accounts may have cost far fewer than forty crackings

(c)

- It is **not** reasonable. Eight characters gives a keyspace small enough that automated tools can search it, and with no complexity requirement the passwords are likely drawn from common words, which a rainbow table covers directly; and because computational processing power keeps improving, a length that was adequate when the policy was written is not adequate now - which is why key- and password-length recommendations are revised over time

(d)

- many users reuse the same password across services, and this is confirmed here - so the adversary can simply try the recovered credentials at the journal service, which will accept them as valid logins without any breach of its own

(e)

- Any four, spread across the categories: **salt** every password before hashing; enable **lockout** after a small number of invalid attempts; require **complexity** and a longer **minimum length**; enforce **password history** and a **maximum age**; deploy **MFA**. Single most effective: **salting the stored hashes** - it is what made the offline attack cheap, and it is the only one of these that protects the passwords once the directory has already been copied. Accept **MFA** with the justification that even correctly cracked passwords would not have granted access