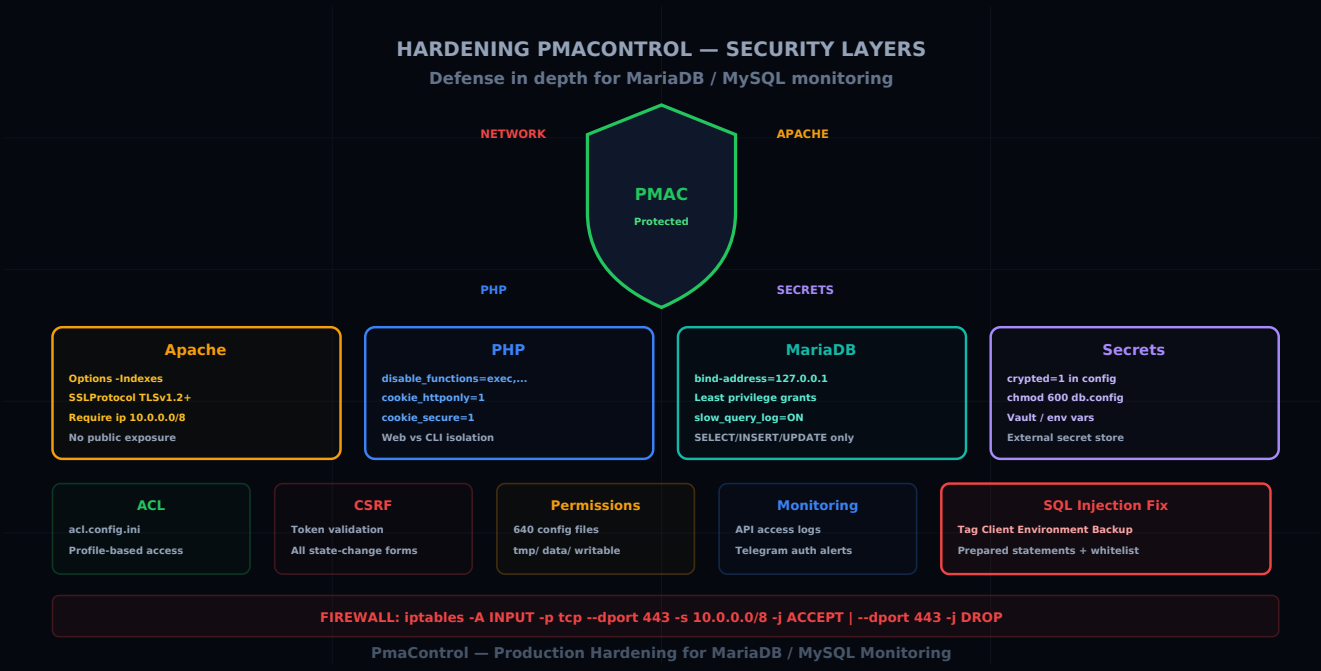
تصلب PmaControl في الإنتاج: دليل الأمان الكامل

Aurélien LEQUOY · 13 أبريل 2026

PMACONTROL SECURITY HARDENING APACHE PHP MARIADB



PmaControl لديه مفاتيح المملكة

PmaControl يشرف على خوادم الإنتاج MariaDB / MySQL. يقوم بتخزين بيانات اعتماد الاتصال ومفاتيح SSH ومقاييس الأداء وبنية قاعدة البيانات. إذا قام أحد المهاجمين باختراق PmaControl، فمن المحتمل أن يكون لديه حق الوصول إلى **البنية الأساسية لقاعدة البيانات بالكامل**.

يوضح هذا الدليل تفاصيل إجراءات التقوية التي يجب تطبيقها قبل وضع PmaControl في الإنتاج. إنه يأتي من تدقيق أمني داخلي ويغطي كل طبقة: Apache، PHP، MariaDB، الأسرار، ACL، CSRF، أذونات الملفات والمراقبة.

الطبقة الأولى: أباتشي

تعطيل قائمة الدليل

افتراضيًا، يمكن لـ Apache عرض محتويات الدلائل بدون ملف فهرس. هذا تسرب للمعلومات:

```
<Directory /srv/www/pmacontrol>
    Options -Indexes
    AllowOverride All
    Require all granted
</Directory>
`
```

-Indexes` غير قابل للتفاوض. بدونها، يمكن للمهاجم استكشاف بنية المشروع والعثور على ملفات التكوين والسجلات والمقالب.

فرض HTTPS

PmaControl ينقل بيانات الاعتماد بنص واضح في طلبات HTTP. بدون HTTPS، يمكن للمهاجم على الشبكة اعتراضها:

```
<VirtualHost *:80>
    ServerName pmacontrol.internal.company.com
    Redirect permanent / https://pmacontrol.internal.company.com/
</VirtualHost>

<VirtualHost *:443>
    ServerName pmacontrol.internal.company.com
    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/pmacontrol.pem
    SSLCertificateKeyFile /etc/ssl/private/pmacontrol.key

    # Modern TLS only
    SSLProtocol -all +TLSv1.2 +TLSv1.3
    SSLCipherSuite HIGH:!aNULL:!MD5:!3DES

    DocumentRoot /srv/www/pmacontrol
</VirtualHost>
```

يقتصر على الشبكة الداخلية

PmaControl لا ينبغي **أبدًا** أن يتم عرضه على الإنترنت. تقييد الوصول إلى الشبكة الداخلية:

```
<Location />
    Require ip 10.0.0.0/8
    Require ip 172.16.0.0/12
    Require ip 192.168.0.0/16
</Location>
```

أو الأفضل من ذلك: ضع PmaControl خلف شبكة VPN والا تكشفه على الإطلاق عبر Apache العام.

إزالة المضيف الافتراضي

يستجيب مضيف Apache الافتراضي (`000-default.conf`) لأي استعلام على عنوان IP الخاص بالخادم. حذفه:

```
a2dissite 000-default.conf
systemctl reload apache2
```

رؤوس الأمان

أضف رؤوس أمان HTTP:

```
Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
Header always set X-XSS-Protection "1; mode=block"
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline'; style-src 'self' 'unsafe-inline'"
```

الطبقة الثانية: PHP

تعطيل الوظائف الخطرة

يستخدم PmaControl `exec()` و `shell_exec()` لعمليات معينة (SSH، المجموعة). والحل ال يكمن في تعطيلها على مستوى العالم، بل في عزل العمال الذين يحتاجون إليها.

بالنسبة لمضيف الويب (الواجهة):

```
; php.ini ou .user.ini dans le DocumentRoot
disable_functions = exec,shell_exec,system,passthru,popen,proc_open
```

```
expose_php = Off
```

للعاملين في CLI (المكنسة الكهربائية، المستمع):

```
; php-cli.ini — ces workers ont besoin de shell_exec
disable_functions =
```

ويضمن هذا الفصل عدم قدرة واجهة الويب على تنفيذ أوامر النظام، حتى لو وجد المهاجم ثغرة أمنية.

جلسات آمنة

```
session.cookie_httponly = 1
session.cookie_secure = 1
session.cookie_samesite = Strict
session.use_strict_mode = 1
session.name = PMACSESSID
`
```

cookie_httponly `يمنع JavaScript من الوصول إلى ملف تعريف ارتباط الجلسة (حماية XSS).`
cookie_secure `يفرض الإرسال عبر HTTPS فقط.` cookie_samesite = Strict` يحمي من CSRF الأساسي.

الحد من التحميل والتنفيذ

```
upload_max_filesize = 2M
post_max_size = 8M
max_execution_time = 30
max_input_time = 60
memory_limit = 256M
```

PmaControl لا يحتاج إلى تحميلات ضخمة. الحد لتقليل سطح الهجوم.

إخفاء نسخة PHP

```
expose_php = Off
```

يؤدي هذا إلى إزالة رأس `X-Powered-By: PHP/8.x` من استجابات HTTP.

الطبقة 3: MariaDB

تقييد امتيازات المستخدم PmaControl

بعد التثبيت، غالبًا ما يتمتع المستخدم PmaControl بالامتيازات الكاملة. تقييده:

```
-- Révoquer les privilèges excessifs
REVOKE ALL PRIVILEGES ON *.* FROM 'pmacontrol'@'localhost';

-- Accorder uniquement ce qui est nécessaire
GRANT SELECT, INSERT, UPDATE, DELETE ON pmacontrol.* TO 'pmacontrol'@'localhost';
GRANT SELECT ON performance_schema.* TO 'pmacontrol'@'localhost';
GRANT REPLICATION CLIENT ON *.* TO 'pmacontrol'@'localhost';
GRANT PROCESS ON *.* TO 'pmacontrol'@'localhost';

FLUSH PRIVILEGES;
```

مبدأ الامتياز الأقل: PmaControl يحتاج فقط إلى قراءة المقاييس والكتابة إلى قاعدته الخاصة.

ربط بالمضيف المحلي

يجب أن تستمع قاعدة البيانات PmaControl فقط على الواجهة المحلية:

```
[mysqld]
bind-address = 127.0.0.1
```

إذا كان PmaControl وقاعدته موجودين على نفس الخادم (التكوين النموذجي)، فلا يوجد سبب للاستماع عبر الشبكة.

تمكين تسجيل الطلبات الحساسة

```
[mysqld]
general_log = OFF          # Trop verbeux en production
slow_query_log = ON
long_query_time = 1
log_error = /var/log/mysql/error.log
```

يساعد سجل الاستعلام البطيء في اكتشاف الاستعلامات غير الطبيعية التي قد تشير إلى حقنة SQL مستغلة.

الطبقة الرابعة: الأسرار

تشفير بيانات الاعتماد

PmaControl يخزن بيانات اعتماد تسجيل الدخول في `db.config.ini.php`. هذا الملف يدعم التشفير:

```
; configuration/db.config.ini.php
[default]
driver = mysql
host = 127.0.0.1
port = 3306
login = pmacontrol
password = "ENCRYPTED_VALUE_HERE"
database = pmacontrol
crypted = 1
```

تخبر العلامة `crypted=1` PmaControl بفك تشفير كلمة المرور في وقت التشغيل. مفتاح التشفير منفصل عن ملف التكوين.

استخدم مخزناً سرياً خارجياً

بالنسبة لعمليات نشر الإنتاج المهمة، قم بالاستعانة بمصادر خارجية للأسرار:

- **Vault** (HashiCorp): يستطيع PmaControl قراءة الأسرار عبر API
- **AWS Secrets Manager** أو **GCP Secret Manager**: لعمليات النشر السحابية
- **متغيرات البيئة**: الحد الأدنى القابل للتطبيق، أفضل من النص العادي

```
# Exemple avec variables d'environnement
export PMAC_DB_PASSWORD="secret_value"
export PMAC_SSH_PASSPHRASE="ssh_secret"
```

حماية ملفات التكوين

```
# Propriétaire : www-data (l'utilisateur Apache)
chown root:www-data /srv/www/pmacontrol/configuration/*.php

# Permissions : lecture pour le groupe, rien pour les autres
chmod 640 /srv/www/pmacontrol/configuration/*.php

# Le fichier de credentials ne doit être lisible que par www-data
chmod 600 /srv/www/pmacontrol/configuration/db.config.ini.php
```

الطبقة الخامسة: ACL (قوائم التحكم في الوصول)

مرجعة acl.config.ini

يحتوي PmaControl على نظام ACL قائم على ملف التعريف. يحدد الملف `acl.config.ini` ملف التعريف الذي يمكنه الوصول إلى وحدة التحكم.

```
; configuration/acl.config.ini
[admin]
* = allow

[dba]
Slave = allow
Server = allow
Dashboard = allow
Backup = deny
Config = deny

[readonly]
Slave = allow
Server = allow(show)
Dashboard = allow
* = deny
```

القواعد الأساسية:

- **تقييد وحدات التحكم الحساسة**: `Config`، `Backup`، `Install`، `Api` يجب أن يكون الوصول إليها متاحًا للمسؤولين فقط
- **إنشاء ملف تعريف للقراءة فقط**: للمطورين الذين يحتاجون إلى استشارة دون تعديل
- **التدقيق بانتظام**: تأكد من أن وحدات التحكم الجديدة المضافة مغطاة بقوائم ACL

حماية نقاط النهاية الهامة

بعض نقاط النهاية حساسة بشكل خاص:

```
[admin]
Install = allow      ; Installation / réinstallation
Config = allow       ; Modification de la configuration
Api = allow          ; API REST complète
Backup = allow       ; Accès aux backups (contient des données)
```

```
[dba]
Install = deny       ; JAMAIS accessible aux non-admins
Config = deny
Api = allow(read)    ; Lecture seule via API
Backup = deny
```

الطبقة السادسة: CSRF (تزوير الطلبات عبر المواقع)

التحقق من وجود الرموز المميزة

يجب أن يتضمن كل نموذج PmaControl رمز CSRF المميز:

```
<form method="POST" action="/slave/start/42/">
    <input type="hidden" name="csrf_token" value="<?= $csrf_token ?>">
    <button type="submit">Start Slave</button>
</form>
```

من جانب الخادم، يجب على وحدة التحكم التحقق من صحة الرمز المميز:

```
if ($_POST['csrf_token'] !== $_SESSION['csrf_token']) {
    throw new SecurityException('Invalid CSRF token');
}
```

إجراءات الحماية كأولوية

الإجراءات التي تعدل الحالة هي الأكثر أهمية:

- بدء / إيقاف الرقيق
- تخطي الخطأ
- إضافة / إزالة الخادم
- تعديل التكوين
- إنشاء / حذف المستخدم

بدون حماية CSRF، يمكن للمهاجم إجبار DBA المتصل على إيقاف النسخ المتماثل لخادم الإنتاج عن طريق إرسال رابط مفخخ إليه.

الطبقة السابعة: أذونات الملف

شجرة الأذونات

```
# Répertoire principal : lisible, pas modifiable
chown -R root:www-data /srv/www/pmacontrol/
chmod -R 750 /srv/www/pmacontrol/

# Répertoires d'écriture : www-data propriétaire
chown -R www-data:www-data /srv/www/pmacontrol/tmp/
chown -R www-data:www-data /srv/www/pmacontrol/data/

# Fichiers PHP : lecture seule pour www-data
find /srv/www/pmacontrol/App/ -name "*.php" -exec chmod 640 {} \;

# Configuration : restrictif
chmod 640 /srv/www/pmacontrol/configuration/*.php
chmod 600 /srv/www/pmacontrol/configuration/db.config.ini.php
```

المبدأ: `www-data` يمكنه قراءة الكود والكتابة إلى `tmp/` و `data/`. لا يمكن تعديل التعليمات البرمجية المصدر أو التكوين.

الطبقة الثامنة: المراقبة الأمنية

الوصول إلى السجل API

يجب تسجيل كل مكالمة إلى API REST باستخدام:

- الطابع الزمني
- مصدر IP
- المستخدم (الرمز المميز)
- نقطة النهاية تسمى
- رمز الاستجابة

```
// Dans le middleware API
$log = sprintf(
    "[%s] %s %s %s → %d",
    date('Y-m-d H:i:s'),
    $_SERVER['REMOTE_ADDR'],
    $user->name,
```

```
    $_SERVER['REQUEST_URI'],
    http_response_code()
);
file_put_contents('/var/log/pmacontrol/api.log', $log . "\n", FILE_APPEND);
```

تنبيهات تيليجرام عند فشل المصادقة

قم بإعداد تنبيه Telegram لكل فشل في الاتصال:

```
if (!$auth->isValid()) {
    Telegram::send(
        "□□ Auth failure on PmaControl\n" .
        "IP: " . $_SERVER['REMOTE_ADDR'] . "\n" .
        "User: " . $_POST['login'] . "\n" .
        "Time: " . date('Y-m-d H:i:s')
    );
}
```

يجب أن تؤدي ثلاث حالات فشل من نفس عنوان IP خلال 5 دقائق إلى حدوث حظر مؤقت.

مراقبة ملفات التكوين

استخدم `inotifywait` أو أداة مشابهة لاكتشاف التغييرات غير المصرح بها:

```
inotifywait -m -r /srv/www/pmacontrol/configuration/ -e modify,create,delete |
while read path action file; do
    echo "[$action] $path$file" >> /var/log/pmacontrol/config_changes.log
    # Envoyer alerte Telegram
done
```

الطبقة 9: الشبكة

قواعد جدار الحماية

```
# Autoriser HTTP/HTTPS uniquement depuis le réseau interne
iptables -A INPUT -p tcp --dport 80  -s 10.0.0.0/8 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -s 10.0.0.0/8 -j ACCEPT
iptables -A INPUT -p tcp --dport 80  -j DROP
iptables -A INPUT -p tcp --dport 443 -j DROP
```

```
# Autoriser MySQL uniquement en localhost
iptables -A INPUT -p tcp --dport 3306 -s 127.0.0.1 -j ACCEPT
iptables -A INPUT -p tcp --dport 3306 -j DROP
```

لا يوجد معرض عام

PmaControl لا ينبغي **أبدًا** أن يكون متاحًا الوصول إليه عبر الإنترنت. حتى مع المصادقة، فإن سطح الهجوم كبير جدًا:

- يتم تخزين بيانات اعتماد الخوادم الخاضعة للإشراف
- يتم تخزين مفاتيح SSH
- تتيح لك الواجهة تنفيذ الإجراءات على خوادم الإنتاج

إذا كان الوصول عن بعد مطلوبًا، فاستخدم VPN (WireGuard، OpenVPN) أو نفق SSH.

الطبقة 10: الحقن SQL – العلاج

حدد التدقيق الداخلي لدينا مخاطر الحقن SQL في أربع وحدات تحكم:

المراقب المالي	خطر	عالج
`Tag.php`	البناء الديناميكي لجملة WHERE	استعلامات ذات معلمات
`Client.php`	التسلسل في المرشحات	استعلامات ذات معلمات
`Environment.php`	الاستيفاء المتغير بالترتيب BY	القائمة البيضاء للعمود
`Backup.php`	معلمة لم يتم إلغاؤها في LIKE	استعلامات ذات معلمات

قبل (الضعيفة):

```
// Tag.php — VULNÉRABLE
$sql = "SELECT * FROM tags WHERE name LIKE '%" . $_GET['search'] . "%'";
$results = $db->query($sql);
```

بعد (آمن):

```
// Tag.php — SÉCURISÉ
$sql = "SELECT * FROM tags WHERE name LIKE ?";
$results = $db->query($sql, ['%' . $_GET['search'] . '%']);
```

بالنسبة لعبارات ORDER BY، فإن القائمة البيضاء هي الحل الآمن الوحيد:

```
$allowed_columns = ['name', 'created_at', 'id'];
$sort = in_array($_GET['sort'], $allowed_columns) ? $_GET['sort'] : 'name';
$sql = "SELECT * FROM tags ORDER BY " . $sort;
```

قائمة مراجعة التصلب

قبل وضع PmaControl في الإنتاج، تحقق من صحة كل نقطة:

- [ ] أباتشي: `-Indexes` معطّل
- [ ] أباتشي: HTTPS القسري
- [ ] أباتشي: وصول مقيد إلى الشبكة الداخلية
- [ ] أباتشي: تمت إزالة المضيف الافتراضي
- [ ] PHP: تم تعطيل الوظائف الخطيرة على مضيف الويب
- [ ] PHP: `session.cookie_httponly = 1`
- [ ] PHP: `session.cookie_secure = 1`
- [ ] PHP: `expose_php = Off`
- [ ] MariaDB: مستخدم يتمتع بالحد الأدنى من الامتيازات
- [ ] MariaDB: `bind-address = 127.0.0.1`
- [ ] الأسرار: بيانات الاعتماد المشفرة (`crypted=1`)
- [ ] ملفات التكوين: الأذونات 640
- [ ] ACL: وحدات التحكم الحساسة مقيدة
- [ ] CSRF: الرموز المميزة لجميع أشكال الإجراءات
- [ ] الأذونات: `tmp/` و `data/` فقط المجلدات القابلة للكتابة
- [ ] المراقبة: سجل الوصول API
- [ ] المراقبة: تنبيهات عند فشل المصادقة
- [ ] الشبكة: جدار الحماية موجود
- [ ] الشبكة: لا يوجد عرض عام
- [ ] SQL: الاستعلامات التي تم تكوينها في العلامة والعميل والبيئة والنسخ الاحتياطي

الخلاصة

إن تأمين PmaControl ليس ترفا - بل هو التزام. تتمتع الأداة بإمكانية الوصول إلى خوادم الإنتاج MariaDB / MySQL، وتخزين بيانات الاعتماد، ويمكنها تنفيذ الأوامر عبر SSH.

تتم عملية التصلب في طبقات: كل طبقة (Apache، PHP، MariaDB، Secrets، ACL، CSRF،أذونات، شبكة) تضيف حاجزًا. إذا تم اختراق إحدى الطبقات، فإن الطبقات الأخرى تبطئ المهاجم.

والخبر السار: كل هذه التدابير قياسية وقابلة للتطبيق في يوم عمل واحد. التكلفة لا تذكر مقارنة بمخاطر المساس بالبنية التحتية لقاعدة البيانات الخاصة بك.